Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

2014-12

Performance analysis and enhancements for the music sub-space direction-finding algorithm in the presence of wideband signals

Straessle, Gregory C.

Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/44676

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

PERFORMANCE ANALYSIS AND ENHANCEMENTS FOR THE MUSIC SUBSPACE DIRECTION-FINDING ALGORITHM IN THE PRESENCE OF WIDEBAND SIGNALS

by

Gregory C. Straessle

December 2014

Thesis Advisor:	Herschel H. Loomis, Jr.
Thesis Co-Advisor:	Frank Kragh

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	12-19-2014	Master's Thesis

4. TITLE AND SUBTITLE

PERFORMANCE ANALYSIS AND ENHANCEMENTS FOR THE MUSIC SUBSPACE DIRECTION-FINDING ALGORITHM IN THE PRESENCE OF WIDEBAND SIGNALS

5. FUNDING NUMBERS

6. AUTHOR(S)

Gregory C. Straessle

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

The collection of signals intelligence via passive direction finding and geolocation of radio frequency signals is of great concern to the military for its contribution to the development of battlespace awareness. Basic subspace direction finding techniques provide a method of determining the direction-of-arrival (DOA) of multiple signals on an array of receivers, but they have an inherent limitation in that they are narrowband by design.

The impact of various signal frequencies, bandwidths, and signal to noise ratios present in the source signals received by a sparse array using the multiple signals classification (MUSIC) subspace direction-finding algorithm are evaluated in this thesis. Additionally, two performance enhancements are presented: one that reduces the MUSIC computational load and one that provides a method of utilizing collector motion to resolve DOA ambiguities.

14. SUBJECT TERMS	15. NUMBER OF PAGES
ambiguity, array, direction finding, geolocation, interferometry, MUSIC, multiple signals classification, sparse array, subspace methods, wideband	123
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

PERFORMANCE ANALYSIS AND ENHANCEMENTS FOR THE MUSIC SUBSPACE DIRECTION-FINDING ALGORITHM IN THE PRESENCE OF WIDEBAND SIGNALS

Gregory C. Straessle
Lieutenant Commander, United States Navy
B.S., U.S. Naval Academy, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2014

Author: Gregory C. Straessle

Approved by: Herschel H. Loomis, Jr.
Thesis Advisor

Frank Kragh
Thesis Co-Advisor

Clark Robertson
Chair, Department of Electical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The collection of signals intelligence via passive direction finding and geolocation of radio frequency signals is of great concern to the military for its contribution to the development of battlespace awareness. Basic subspace direction finding techniques provide a method of determining the direction-of-arrival (DOA) of multiple signals on an array of receivers, but they have an inherent limitation in that they are narrowband by design.

The impact of various signal frequencies, bandwidths, and signal to noise ratios present in the source signals received by a sparse array using the multiple signals classification (MUSIC) subspace direction-finding algorithm are evaluated in this thesis. Additionally, two performance enhancements are presented: one that reduces the MUSIC computational load and one that provides a method of utilizing collector motion to resolve DOA ambiguities.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

AIC Akaike information criterion

AOA angle-of-arrival

AWGN additive white gaussian noise

CAF cross-ambiguity function

DFT discrete Fourier transform

DOA direction-of-arrival

ESPRIT estimation of signal parameters via rotational invariance techniques

FDOA frequency difference-of-arrival

FFT fast Fourier transform

FPGA field programmable gate array

IF intermediate frequency

LFM linear frequency modulation

MDL minimum descriptive length

MUSIC multiple signals classification

QPSK quadrature phase-shift keying

RF radio frequency

SIGINT signals intelligence

SNR signal-to-noise ratio

TDOA time difference-of-arrival

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

Background

Joint Publication 1-02 defines signals intelligence (SIGINT) as intelligence derived from communications, electronic, and foreign instrumentation signals [1]. Direction finding and geolocation are important parts of that intelligence; passive direction finding is the process of estimating an emitter's direction from the receiver's position, and geolocation is then the process of using one or more direction-of-arrival (DOA) results in combination with known information about receiver or emitter motion and relative geometry in order to determine the emitter's position in terms of an earth-fixed coordinate system. For the military, obtaining location information for emitters is a vital part of understanding an adversary's order of battle and force strength.

In the future, the signal environment will only become more dense, and the signals themselves will become lower power and wider bandwidth. These trends mean that signals will become more difficult to detect and more likely to overlap in time and bandwidth. Airborne and space-based collectors are ideal for conducting SIGINT because they provide more area coverage and their motion enables finer location resolution than terrestrial systems, but they have limitations on the size and number of receivers which can be carried. The large coverage area exacerbates the possibility of signal overlap, and the distance can make detection of low power signals difficult. Subspace algorithms provide an ability to resolve multiple overlapping signals, but many are based on a narrowband assumption or require strict receiver antenna array arrangements, which can limit their usefulness [2], [3].

Objective

In order to evaluate its performance with wideband signals and sparse arrays, the multiple signals classification (MUSIC) algorithm of [2] has been implemented and examined. MUSIC was one of the first subspace algorithms to be developed; many modifications and improvements to it have been made since its introduction [4]–[6], but an analysis of the performance of the basic algorithm will provide a baseline reference to which other methods can be compared. Additionally, it is a goal of this thesis to evaluate how wideband the input signals can be before the algorithm breaks down. Two enhancements to the al-

gorithm have also been developed: a method of improving the computational performance of MUSIC without sacrificing resolution, and a method of utilizing collector motion to determine emitter location for an array with ambiguities.

Performance Analysis

For the majority of the analysis, a sparse array with six elements, a large baseline, and no ambiguities was utilized for the collector. For the emitters, variations of three basic signal types were modeled: pure sinusoid, linear frequency modulation (LFM), and quadrature phase-shift keying (QPSK). The evaluation was divided into four areas examining the effects on the MUSIC results: signal center frequency, bandwidth, DOA, and signal-to-noise ratio (SNR).

Signals with center frequency other than the collector's center frequency were found to have a shifted DOA in the MUSIC results. This shift can easily be corrected, however, assuming that an accurate estimation can be made of the source signal center frequencies. A corrected DOA can be determined using the ratio of the collector center frequency to the estimated source signal's center frequency.

Signal bandwidth leads to more complicated distortions that cannot be as easily corrected as can signal frequency offsets. The first effect to note is that the amplitude of the MUSIC response decreases with increased bandwidth, though even at bandwidths up to 40 percent of the center frequency a peak of sufficient amplitude above the background can be resolved. In testing signals with two widely separated DOAs but significantly overlap in bandwidth, it was found that MUSIC was easily able to resolve the two signals. A third test was conducted to determine the performance for wideband signals with close DOAs but separation in frequency, and MUSIC was again able to resolve the DOAs of two signals. In this test, however, the signals were imperfectly reconstructed with blending of the two signals present which became worse with an increase in bandwidth.

A final bandwidth analysis with a third signal which was separated in DOA but overlapped in frequency with one of the prior two was then evaluated. It was found that when the overlap is slight (narrow bandwidth third signal), the DOAs are all able to be resolved via MUSIC, but when the overlap is large (wide bandwidth third signal), the MUSIC response becomes degraded: the two closely spaced signals can no longer be resolved and

the third signal response can be distorted.

The effect of DOA on the MUSIC response is related to the bandwidth. For sinusoidal signals, the DOA has little to no impact on the MUSIC response, but for wideband signals the amplitude of the MUSIC peak can widen and decrease drastically as the offset from array boresight increases.

The last signal parameter evaluated was the effect of SNR. One might expect that the MUSIC peak amplitudes would decrease with decreased SNR, and this was found to be the case. An examination of the statistics of the Monte Carlo simulation reveals more information, at low SNRs the mean estimated DOA is still accurate, but the standard deviation increases drastically. A second evaluation of SNR effects was conducted with two closely spaced signals, and it was found that the signal DOAs were easily resolvable at 10 dB SNR and just resolvable at 0 dB SNR, but at -10 dB SNR there was only a single peak present. A final analysis examined the impact of multiple signals with different SNRs. Here bandwidth played a role again in leading to additional distortions beyond those of narrowband signals. In general, sinusoidal signals could be still be resolved up to a 20 dB SNR difference, but signals with 10% bandwidth could only be resolved up to a 10 dB difference in SNR.

Computational Enhancement

One of the major limitations of the MUSIC algorithm is that the search over the angular visible range is computationally expensive. The primary driver of that computation is not the MUSIC calculation itself, but the fact that the MUSIC calculation must be performed over a discrete range of possible signal DOAs. In order to obtain a high resolution estimate of the DOA a small angular step size must be used which means a large number of discrete angles must be evaluated. A method of reducing the number of discrete angles without also reducing the resolution would be to run the MUSIC algorithm more than once. On the first run, a large step size should be used which provides only a rough estimate of signal direction. Those estimates can then be used to re-run MUSIC with a much smaller step size over only the angles surrounding each of the signal direction estimates, with the result of obtaining the high resolution DOA estimate desired. With two signals present, it was found that a $\sim 30\times$ improvement could be obtained on an azimuth-only evaluation with a

final resolution of approximately 100 ft at a distance of nearly 200 nmi.

Ambiguity Resolution

Array ambiguities, which arise when the minimum receive element spacing is too large, are unresolvable for fixed arrays using subspace methods. Ambiguity resolution can be accomplished, however, when the collector is moving at a known velocity and multiple collections of the same signal can be obtained. This is because the true DOA will always point in the direction of the emitter, but any ambiguous DOAs will tend to drift. By tracking the intersection points of DOA vectors from subsequent collections, it should easily be determinable that one set of crossing points remains relatively fixed while another set drifts. Based on this, a metric was developed which uses the motion of subsequent DOA intersection points and can provide an estimate of the correct DOA in only three collections.

It was found that this metric was less accurate for wideband and low SNR signals because in those cases the estimated DOAs of each collection were less accurate due to the distortions discussed in the section on performance analysis. Based on an observation that the DOA intersection points of the true DOAs tended to jitter around the true locations, a more robust metric which uses the center point of two subsequent DOA intersections was developed; this method can provide an estimate of the correct DOA in four collections.

Conclusion

It is hoped that the results of this thesis, which are a first step towards deeper investigations into the subspace direction-finding algorithms, will enable an expansion of the geolocation research which has been on-going at the Naval Postgraduate School to include more analysis beyond the more traditional two-receive element geolocation methods. The ability to accurately utilize MUSIC in moderate to low bandwidth environments, despite it being derived specifically for sinusoidal signals, was also demonstrated in this thesis. Additionally, the density of radio frequency (RF) emitters is increasing around the world, and with that comes an increase in the likelihood of receiving overlapping signals. Methods of accurately resolving and geolocation those overlapping signals are likely to be of interest to the military in the future, which means that now is the time for research into subspace direction finding and other promising methods of multiple signal geolocation.

List of References

[1] *Department of Defense Dictionary of Military and Associated Terms*, Joint Publication 1-02, U.S. Joint Chiefs of Staff, Washington, DC, 8 Nov. 2010, (As amended through 15 Aug. 2014).

[2] R. Schmidt, "Multiple emitter location and signal parameter estimation," *Antennas and Propagation, IEEE Transactions on*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[3] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 984–995, July 1989.

[4] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '83.*, vol. 8, Apr 1983, pp. 336–339.

[5] B. Friedlander and A. Weiss, "Direction finding for wide-band signals using an interpolated array," *IEEE Transactions on Signal Processing*, vol. 41, no. 4, pp. 1618–1634, Apr. 1993.

[6] M. Doron, E. Doron, and A. Weiss, "Coherent wide-band processing for arbitrary array geometry," *IEEE Transactions on Signal Processing*, vol. 41, no. 1, pp. 414–417, Jan. 1993.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

I would like to start by thanking my advisors, Professor Herschel Loomis and Professor Frank Kragh, for their support and direction as I worked to gain an understanding of the material and determine where I might, in some small way, be able to add to the knowledge base of those who came before me. Your advice has always been exactly what I needed to focus my efforts.

To the faculty, staff, and fellow students of the Space Systems Academic Group, I would like to say a sincere thank you for broadening the horizons of this lowly helicopter pilot and helping to prepare me for working in the field of space acquisitions.

Lastly but most importantly I would like to thank my wife for her unceasing love and support. You have continually and selflessly picked up responsibilities at home that I have had to neglect to focus more and more time on classes, projects, and this thesis. I thank God every day for bringing you into my life!

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

1.1 Background

With radio-frequency signals, one is frequently only concerned with the signal content, or message. There are numerous situations and signal types, however, where descriptive information about the signal may be more useful. With radar, the timing and doppler return of a signal provide information about a target. With beacons, the desire is to determine one's position with respect to one or more fixed emitters, or an emitter's location with respect to a known position. For the military, Joint Publication 1-02 defines signals intelligence (SIGINT) as intelligence derived from communications, electronic, and foreign instrumentation signals [1]. By examining received radar pulses, determinations can be made about the type of radar, its location, and sometimes even the specific piece of equipment; similarly, through evaluation of received communication signals, determinations can be made about an emitter's location and sometimes about the content being transmitted. Passive direction finding is the process of estimating an emitter's direction from the receiver's position, and geolocation is then the process of using one or more direction-of-arrival (DOA) results in combination with known information about receiver or emitter motion and relative geometry in order to determine the emitter's position in terms of an earth-fixed coordinate system. For the military, obtaining location information for emitters is a vital part of understanding an adversary's order of battle and force strength.

In detecting a signal and obtaining an accurate geolocation, numerous complications can arise. Low power and short duration signals are both more difficult to detect. Additionally, the task of geolocation is complicated when multiple signals overlap in time, frequency, or direction. Airborne and space-based collectors are ideal for conducting SIGINT because they provide more area coverage and their motion enables finer location resolution than terrestrial systems, but they have limitations on the size and number of receivers which can be carried. While the time overlap of short-duration signals are generally unlikely, the likelihood increases with increased collector coverage area and in particularly signal-dense environments. Many of the current algorithms for separating overlapping signals are based

on a narrowband assumption or require strict receiver antenna array arrangements, which can limit their usefulness [2], [3].

1.2 Objectives

The implementation and performance of the multiple signals classification (MUSIC) subspace direction-finding algorithm with a sparse receive antenna array is examined in this thesis. MUSIC is based on a narrowband assumption, and it is a goal of this thesis to evaluate how wideband the input signals can be before the algorithm breaks down. Basic representative types of narrowband and wideband signals are utilized which may overlap in time, frequency, and direction and the impact of those overlaps and interactions on the ability to accurately determine an emitter's DOA is evaluated. Specific focus is made on the effects of signal frequency, bandwidth, DOA, and signal-to-noise ratio (SNR) on the MUSIC output. Additionally, a method of improving the computational performance of MUSIC without sacrificing accuracy is presented. Finally, a method of utilizing collector motion to determine emitter location in the presence of multiple ambiguous DOAs is developed for the case of a sparse array with inherent ambiguities.

This analysis was conducted solely as a software simulation written in MATLAB. In addition to implementation of the MUSIC algorithm, test signal generation, a least-squares emitter location estimation algorithm, and a signal reconstruction algorithm based on the estimated DOAs have also been implemented. The code is included in Appendix B for the reader to reference or utilize for future work.

1.3 Related Work

The mathematics behind traditional passive direction finding and geolocation have been presented in numerous papers and reports. Specifically for this thesis, the clear presentation of the methods of geolocation for two receiver elements in reports by Professor Herschel Loomis [4] and Michael Price [5] were invaluable to gaining an understanding of the basic direction finding and geolocation processes and the uncertainties involved.

In order to determine the DOA of multiple simultaneous signals, a receiver antenna array with three or more receive elements is required. There are two primary methods of performing the analysis of such an array—maximum-likelihood and signal subspace

methods. The maximum-likelihood estimator is based on a minimization of the likelihood function, which is a non-linear optimization problem and is generally computationally intensive to calculate, while subspace methods are based on the eigen-decomposition of the signal covariance matrix, for which an estimate is relatively simple to compute [6]. The MUSIC method of direction finding developed by Schmidt [2] was one of the first subspace methods proposed; since its proposal there have been multiple modifications to the MUSIC algorithm to improve resolution or reduce computational complexity, most notably Barabell's root-MUSIC [7]. The advantages of the original MUSIC algorithm are that it is relatively simple to understand and can be applied to receiver arrays of arbitrary spacing, though it is still computationally intensive because it relies on a search across all possible signal DOAs. Root-MUSIC avoids this search by examining the roots of the spectrum polynomial, but is only applicable for uniformly spaced arrays. Another method proposed by Roy and Kailath called estimation of signal parameters via rotational invariance techniques (ESPRIT) [3] is a popular subspace method which is contingent on a matched pair of arrays of which the sensors all have identical displacement vectors; like root-MUSIC it also avoids a computationally intensive search, but it is a poorer choice for aerospace applications due to the requirement for twice as many receivers as the original MUSIC algorithm.

Two drawbacks to all of the subspace methods listed above are that they are non-coherent, meaning that they will not work if two or more of the received signals are highly correlated, and they are designed to work with narrowband signals. The first drawback can be overcome via application of spatial smoothing [8], [9]. The second drawback has led to the development of various alternative processing methods. Two similar methods which derive a composite covariance matrix through a combination of interpolations of the array manifold for different frequency bands were developed by Weiss, Doron, and Friedlander [10], [11]. Any of the subspace methods listed in the previous paragraph can then be applied to this modified covariance matrix; these methods also have the added advantage of being coherent and able to handle correlated signals.

In terms of performance analysis, Kaveh and Barabell [12] conducted an in-depth statistical analysis of the resolution threshold with respect to SNR of two very closely spaced narrowband plane waves, and Swindlehurst and Kailath [13] investigated the performance

of MUSIC in light of the uncertainties inherent in estimation of the array covariance matrix and the array manifold. An analysis of the performance of a sparse array in the presence of wideband signals of varying frequencies is not readily apparent in the literature, though an understanding of these cases is useful for the implementation of a subspace algorithm in an airborne or spaceborne environment and for being able to make informed decisions about appropriate selection of a wideband correction method, if one is required at all for the target signals-of-interest.

1.4 Thesis Organization

Some background information on the importance of emitter location, especially for the military, was provided in this chapter. The thesis objective was also presented as was related research on which this thesis builds.

The mathematical algorithms upon which the thesis work relies is discussed in Chapter 2, Emitter Location Processing. First presented are the basic types of direction finding processing with one or two receive antennas, followed by a discussion of geolocation methods. The MUSIC algorithm and the underlying concepts of the matrix pseudo-inverse, eigen-decompositions, and covariance matrices are then detailed. A subspace visual depiction of the MUSIC algorithm is also provided to assist with understanding the algorithm from a geometric perspective.

In Chapter 3, MUSIC Performance Analysis, an in-depth analysis of the MUSIC algorithm's response to variations in the frequency, bandwidth, DOA, and SNR of a set of standardized test signals is provided. In particular, situations where the algorithm breaks down and the results are degraded or distorted are pursued and analyzed for their most likely causes.

In Chapter 4, MUSIC Performance Enhancements and Limitations, two enhancements to the MUSIC algorithm and some of the limitations inherent in a real-world implementation are provided. The first enhancement is a method of reducing the computational complexity without loss of resolution, and the second is a method of using collector motion to resolve array ambiguities.

In Chapter 5, Conclusion, the analysis conducted, its significance, and recommendations

for future work are discussed.

In Appendix A, Subspace Processing Specifics, the differences and similarities of time-domain and frequency-domain methods of subspace processing as well as considerations for the implementation of MUSIC processing at intermediate frequencies are discussed. In Appendix B, MATLAB Code, the MATLAB code used to implement the MUSIC algorithm and conduct emitter geolocation is provided.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Emitter Location Processing

2.1 Basic Direction Finding

There are four primary methods of passively determining the DOA of signals when only one or two receivers are utilized: angle-of-arrival (AOA), interferometry, time difference-of-arrival (TDOA), and frequency difference-of-arrival (FDOA). The following paragraphs will expand on the processing methodologies for each in two dimensions; the expansion to three dimensions is straightforward. The methods are listed below in order of increasing processing complexity.

AOA requires the least amount of signal processing, but it is contingent on the use of a high-gain, narrow-beamwidth receiver. The estimated signal DOA is simply the direction in which the receiver is pointed, or scanned in the case of a phased array antenna. AOA is also unique among the direction-finding methods for only requiring one receiver; it is historically referred to as radio direction finding [4].

Interferometry could also be called phase difference-of-arrival, because the phase difference of the signal incident on two or more receiving elements is exploited to determine the DOA [5]. A depiction of the two-receive element case in two dimensions, where the distance between receive elements (R_1, R_2) in the array is d, is provided in Fig. 1. The emitter (E) location is defined in terms of polar coordinates with distance r and angle from receiver array broadside θ, which is defined from a common origin point of the array. Additionally, the emitter is assumed to be in the far field ($r >> d$) with the signal arriving at the receivers as a plane wave. Unless the emitter is directly broadside to the receivers ($\theta = 0^\circ$), this plane wave will arrive at one of the receivers before the other, resulting in a phase difference ϕ at each of the receiving elements. The path difference δ which the wave travels to reach the second receiver is related to the DOA θ and the receiver separation d by the relationship $\delta = d \sin\theta$. From this, the phase difference can be determined as

$$\phi = k\delta = 2\pi \frac{d}{\lambda} \sin\theta, \tag{2.1}$$

where the wavenumber $k = 2\pi/\lambda$ and λ is the wavelength; (2.1) also assumes that the signal frequency and wavelength do not change during the additional transit time. Determining the DOA is then simply a matter of solving for θ in (2.1) [5]. One of the primary drawbacks of this method is that since the phase is cyclical over $\phi \in [-\pi, \pi)$, the maximum unambiguous spacing between elements over $\theta \in [-90^\circ, 90^\circ]$ is $d = \lambda/2$. At separations larger than this, the number of ambiguities present will depend on d, λ, and θ.

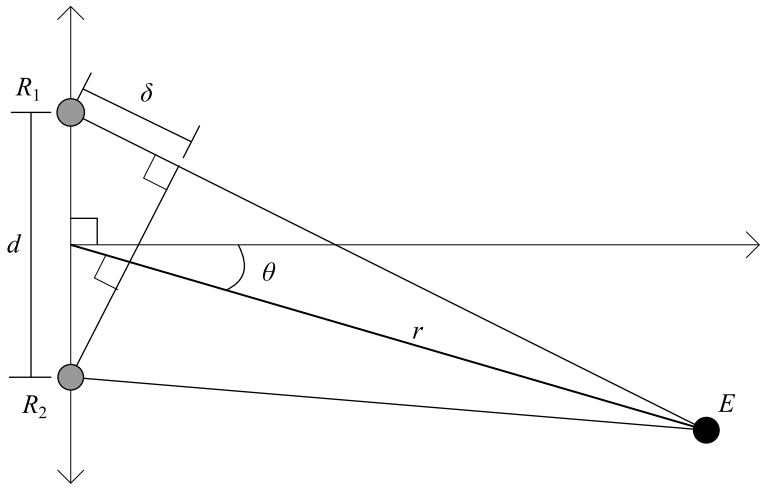


Figure 1. Two-element interferometer, after Ref. [5].

TDOA takes advantage of the same difference in path length as interferometry except that the time difference τ is being utilized vice the phase difference. For TDOA, the path difference is defined as $\delta = c\tau$. Solving for θ directly as was done in the interferometric case limits τ to a narrow interval in order to avoid phase ambiguities; instead one can treat the emitter's location as an unknown position (x_e, y_e) and express the path difference as [4]

$$\delta = \sqrt{\left(x_e + \frac{d}{2}\right)^2 + y_e^2} - \sqrt{\left(x_e - \frac{d}{2}\right)^2 + y_e^2}, \tag{2.2}$$

assuming that the receivers are equidistant from the origin and lie on the x-axis. Equation (2.2) can be re-arranged to the form [4]

$$1 = \frac{x_e^2}{\frac{\delta^2}{4}} - \frac{y_e^2}{\frac{d^2 - \delta^2}{4}}, \tag{2.3}$$

which is recognizable as the formula for a hyperbola. This hyperbola, known as an isochrone, connects all the possible emitter locations which result in the same TDOA; the proper leg of the hyperbola over $\theta \in (-90°, 90°)$ can be determined by the sign of the path difference. An example of a set of isochrones is presented in Fig. 2. In this manner, the only practical limit to receiver spacing lies in ensuring that both receivers can still receive the same signal wavefront at approximately the same power, and receivers can thus be spaced much further apart than $\lambda/2$. The TDOA method of direction finding is generally only applied by itself to signals which have characteristics enabling precise time-of-arrival determination such as radar pulses.

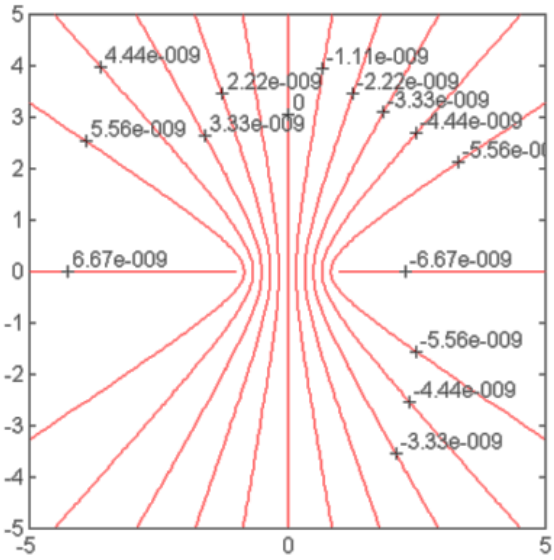


Figure 2. Example set of TDOA isochrones, from Ref. [4].

Rather than being based on the path difference, FDOA arises from an artifact of receiver motion during signal collection – Doppler effect. Since the two receivers have different velocities relative to the emitter, a difference in Doppler frequency can be determined and exploited to assist in geolocation [5]. Making the assumption that the emitter is not moving, we find that the Doppler shift at one receiver is the product of the frequency with the ratio of the closing velocity with the speed of light: $\Delta f = f v_{closing}/c$. The FDOA γ is then the Doppler difference between the two emitters, $\gamma = f(v_{closing,1} - v_{closing,2})/c$. Since the closing velocity can be expressed as the dot product of the receiver velocity and the unit

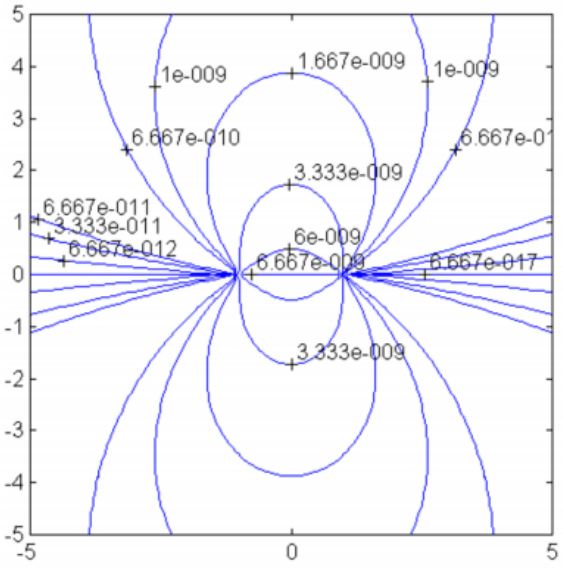


Figure 3. Example set of FDOA isodops, from Ref. [4].

vector in the direction of the emitter, in the x-y plane the FDOA can be expressed as [4]

$$\gamma = \frac{f}{c}\left(\frac{v_x(x_e+s)+v_y y_e}{\sqrt{(x_e+s)^2+y_e^2}} - \frac{v_x(x_e-s)+v_y y_e}{\sqrt{(x_e-s)^2+y_e^2}}\right), \tag{2.4}$$

under the assumption that the receivers have the same velocity vector and are both on the x-axis at distance s from the origin, and where v_x and v_y are the closing velocities in the x- and y-directions. The curves connecting points of equal FDOA are termed isodops. While not obvious from (2.4), this equation defines a leminscate, or "two-leafed rose," when the receivers are moving along the x-axis, which is illustrated by the example set of isodops in Fig. 3; when the receivers are moving along the y-axis, a "four-leafed rose" shape arises [5]. While a single FDOA isodop does not provide an indication of signal DOA like a single isochrone from TDOA or interferometry, it is highly useful in improving the position accuracy when used with an isochrone.

2.2 The Cross-Ambiguity Function

When either the transmitter or receivers are moving, the Doppler difference in the received signals prevents the determination of TDOA via direct correlation, and the cross-ambiguity function (CAF) must be used to estimate both TDOA and FDOA simultaneously. The

generalized form of the CAF is [14]

$$AF(\tau,\gamma) = \int_0^T s_1(t)s_2^*(t+\tau)e^{-j2\pi\gamma t}\,dt, \tag{2.5}$$

where s_1 and s_2 are the analytic signals at the two receivers, * indicates the complex conjugate, and τ and γ are the possible TDOA and FDOA values over which the analysis is conducted. The estimated TDOA and FDOA of the collected signal will be where $|AF(\tau,\gamma)|$ has a peak [14]. The TDOA and FDOA results of the CAF improve in resolution with increases in the signal bandwidth and the collection duration, respectively; thus, the relatively long collection durations possible on continuous wave signals also make them prime candidates for analysis via CAF due to the fine FDOA resolution [14].

2.3 Position Estimation

A set of DOA estimates obtained from any of the methods above can be combined to estimate the emitter's location. In three-dimensional space, a minimum of three DOAs is required for a position estimate, though two can be used with an assumption that the emitter is located on the surface of the earth. Use of more DOA measurements will result in an over-constrained solution; however, due to noise and measurement and timing inaccuracies, the DOAs will not all intersect at a single point. In such a case a least-squares solution is desired, which determines the point of minimum residue. Since their defining equations ((2.3), (2.4)) are non-linear, the traditional solution using TDOA, FDOA, or CAF results is to derive a linear approximation in the vicinity of an estimated position and use an iterative method such as Newton-Raphson to converge on the local minima [4].

2.4 Subspace Methods of Direction Finding

When the possibility exists of q signals being simultaneously incident on a collector, the individual signal DOAs generally cannot be determined unless the collector employs an array of p receivers, where $p > q$. The previously discussed direction-finding methods are designed for use on a single signal incident on two receivers and cannot be directly applied; therefore, alternative methods which can simultaneously process information from all p receivers are desirable. The subspace direction-finding methods comprise one set of these alternative techniques. For a generic situation with q signals and p receivers, the

sum-of-signals at each receiver can be viewed as the vector multiplication problem

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{w}, \tag{2.6}$$

where $\mathbf{x}$ is the p-length vector of received signals, $\mathbf{A}$ is the signal steering matrix (often referred to as the array manifold), $\mathbf{s}$ is the q-length vector of source signals, and $\mathbf{w}$ is the p-length additive noise process [15]. For the azimuth-only case, (2.6) can be expanded as

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_p(t) \end{bmatrix} = \begin{bmatrix} a_1(\theta_1) & a_1(\theta_2) & \cdots & a_1(\theta_q) \\ a_2(\theta_1) & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_p(\theta_1) & \cdots & \cdots & a_p(\theta_q) \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_q(t) \end{bmatrix} + \begin{bmatrix} w_1(t) \\ w_2(t) \\ \vdots \\ w_p(t) \end{bmatrix} \tag{2.7}$$

to highlight the structure of the steering matrix, where $a_j(\theta_i)$ is the phase shift of the ith source signal at the jth receiver. It should be noted that while (2.7) is expressed as a time-domain relationship, it can just as easily be expressed in the frequency domain. Additionally, the math is simplified by working with analytic signals; thus, if an arbitrary signal $s_i(t)$ is a sine wave at frequency f_i, it would be represented as $s_i(t) = \mathrm{Re}\left\{ e^{j(2\pi f_i t - \pi/2)} \right\}$. Since the real portion of all of the signals is utilized, it is simpler to drop the $\mathrm{Re}\{\}$ notation and work with just the complex representation, which is known as the analytic signal. In this situation, the $a(\theta_i)$ columns of the steering matrix, known as the steering vectors, are a p-length vector of the phase shifts applied to each signal relative to the distance of the receive elements from a common origin:

$$\mathbf{a}(\theta_i) = \begin{bmatrix} e^{j\phi_{i1}} & e^{j\phi_{i2}} & \cdots & e^{j\phi_{ip}} \end{bmatrix}^T, \tag{2.8}$$

where $\phi_{ij} = kd_j \sin(\theta_i)$ for a linear array as defined in (2.1), and d_j is the distance from the jth receive element to the origin. The phase shift is positive for positive d_j and θ because the signal at the jth emitter leads the signal at the origin; in other words the wavefront arrives at the jth emitter prior to arriving at the origin. The $a(\theta)$ steering vectors serve to map the signals into p-dimensional space, and the range of $\mathbf{x}$ will be limited by the range of $\mathbf{A}$, neglecting the effects of the additive noise $\mathbf{w}$. Given that $\mathbf{x}$ and $\mathbf{A}$ are known and $\mathbf{s}$ is desired, since $p > q$ the problem is over-determined and a least-squares solution can

generally be found, understanding that the result will be noisy since $\mathbf{w}$ is unknown. Since θ (and, thus, $\mathbf{A}$) and $\mathbf{s}$ are both unknown, a direct least-squares problem cannot be solved.

Subspace direction-finding methods solve this problem by exploiting the eigen-decomposition of the array spatial covariance matrix [15]. In general, the covariance function between times t_1 and t_2 of two signals $x(t)$ and $y(t)$ which are modeled as random processes is defined as [6]

$$C_{xy}(t_1,t_2) = E\left\{[x(t_1) - E\{x(t_1)\}][y(t_2) - E\{y(t_2)\}]^*\right\}, \tag{2.9}$$

where $E\{\}$ is the expectation operator. The covariance of the same zero mean signal $x(t)$ with itself collapses to the autocorrelation function,

$$R_{xx}(t_1,t_2) = E\{x(t_1)x^*(t_2)\}. \tag{2.10}$$

Since the signals of interest in this direction-finding problem are sums of sinusoids, they are by definition zero-mean. In matrix form over the received signal vector $\mathbf{x}$ then, the covariance matrix of zero-mean signals can be defined as

$$\mathbf{R_{xx}} = E\left\{\mathbf{x}\mathbf{x}^H\right\}, \tag{2.11}$$

where ^{H} represents the conjugate transpose, also known as the Hermitian operator. It should be noted at this point that C and R are traditionally used by many textbooks to denote the covariance and correlation functions, respectively. Most of the subspace method literature appears to use the $\mathbf{R}$ notation but defines it as the array covariance matrix; since zero-mean signals are generally assumed, the covariance and correlation matrices will be equal, but the notation can still be slightly confusing. The convention utilized by previous subspace papers is used in this thesis. In practice, $\mathbf{R_{xx}}$ cannot be determined exactly since the signal information is not known for all time, but an estimate can be made over a collected sample set as [6]

$$\mathbf{R_{XX}} = \frac{1}{N}\mathbf{X}\mathbf{X}^H, \tag{2.12}$$

where the rows of $\mathbf{X}$ are the collected samples, making $\mathbf{X}$ $p \times N$, assuming N samples at each receiver. Substituting (2.6) into (2.11), we derive the $p \times p$ array covariance matrix in

terms of the transmitted signal and noise

$$\begin{aligned} \mathbf{R_{xx}} = E\left\{\mathbf{x}\mathbf{x}^H\right\} = E\left\{(\mathbf{A}\mathbf{s}+\mathbf{w})(\mathbf{A}\mathbf{s}+\mathbf{w})^H\right\} = \\ E\left\{\mathbf{A}\mathbf{s}\mathbf{s}^H\mathbf{A}^H + \mathbf{w}(\mathbf{A}\mathbf{s})^H + (\mathbf{A}\mathbf{s})\mathbf{w}^H + \mathbf{w}\mathbf{w}^H\right\} = \\ E\left\{\mathbf{A}\mathbf{s}\mathbf{s}^H\mathbf{A}^H + \mathbf{w}\mathbf{w}^H\right\} = \mathbf{A}\mathbf{R_{ss}}\mathbf{A}^H + \mathbf{R_{ww}} \end{aligned} \quad , \tag{2.13}$$

where $\mathbf{R_{ss}}$ is the source signal covariance matrix and $\mathbf{R_{ww}}$ is the noise covariance matrix. In this derivation, the assumption is made that $\mathbf{s}$ and $\mathbf{w}$ are uncorrelated. If the further assumption is made that the noise is additive white gaussian noise (AWGN), then $\mathbf{R}_{ww} = \sigma_{\mathbf{w}}^2\mathbf{I}_p$, where $\sigma_{\mathbf{w}}$ is the standard deviation of the noise and $\mathbf{I}_p$ is the $p \times p$ identity matrix. By construction, $\mathbf{R_{ss}}$ is $q \times q$ and rank q; pre- and post- multiplying by $\mathbf{A}$ makes $\mathbf{A}\mathbf{R_{ss}}\mathbf{A}^H$ $p \times p$, but it will still be rank deficient with rank q. The received signal covariance matrix $\mathbf{R_{xx}}$ and the AWGN $\sigma_{\mathbf{w}}\mathbf{I}_p$ are $p \times p$ with rank p, and some conclusions can then be drawn about the eigen-space of each of the matrices.

An eigenvalue/eigenvector pair are a scalar and a vector which satisfy the equation

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}, \tag{2.14}$$

where $\mathbf{A}$ is an arbitrary square matrix, $\mathbf{v}$ is an eigenvector, and λ is the eigenvalue. There are numerous methods of conducting the eigen-decomposition of matrix which can be found in many textbooks and will not be reviewed here. It is often useful to think about eigenvectors in a geometric sense: let $\mathbf{B}$ be a $2 \times N$ set of (x,y) sampled measurements which are randomly distributed over an ellipse, such as a cannon being fired N times and the cannonball's landing position being recorded. The covariance matrix of the data would then be the 2×2 matrix $\mathbf{R_{BB}}$. This matrix will have two 2×1 eigenvectors which will be vectors in the directions of the major and minor axes of the data, and the eigenvalue associated with the semi-major eigenvector will be larger than that associated with the semi-minor eigenvector. Additionally, a covariance matrix is always Hermitian symmetric and positive semi-definite, so the eigenvalues will always be ≥ 0 and the eigenvectors will be orthogonal. The eigenvalues will also be equal to the variance of the original data in that direction [6]. If the set of samples in $\mathbf{B}$ were all exactly along a line, then one of the eigenvalues of $\mathbf{R_{BB}}$ would be zero since the matrix $\mathbf{B}$ has no information in the direction of the eigenvector associated with the zero eigenvalue (i.e., zero variance); since orthogonal

eigenvectors form a multidimensional basis, this means there is no information in one of the dimensions.

Returning to (2.13), since $\mathbf{AR_{SS}A}^H$ is rank q and is positive semi-definite by construction, it will have $p-q$ eigenvalues which are exactly equal to zero since a rank deficiency indicates that there is no information in one or more dimensions. The AWGN, however, notionally exists in all directions equally with p repeated eigenvalues of $\sigma_{\mathbf{w}}^2$. This means that if the eigenvectors of $\sigma_{\mathbf{w}}^2 I_p$ are matched exactly to the eigenvectors of $\mathbf{AR_{ss}A}^H$, the eigenvalues of $\mathbf{R_{xx}}$ will be a combination of the eigenvalues of $\mathbf{AR_{ss}A}^H$ and $\sigma_{\mathbf{w}}^2 I_p$. Since $p-q$ eigenvalues of $\mathbf{AR_{ss}A}^H$ are zero, $p-q$ eigenvalues of $\mathbf{R_{xx}}$ must be equal to $\sigma_{\mathbf{w}}^2$ because the only contribution in those dimensions is the AWGN. Additionally, assuming the SNR is greater than 0 dB, the q non-zero eigenvalues of $\mathbf{AR_{ss}A}^H$ are larger than $\sigma_{\mathbf{w}}^2$ and the contribution from the array covariance matrix will dominate the q largest eigenvalues of $\mathbf{R_{xx}}$. Therefore, the received signal subspace can be thought of as having q signal eigenvectors ($\mathbf{E_s}$) and $p-q$ noise eigenvectors ($\mathbf{E_w}$). This is the basic predicate behind all of the subspace direction-finding methods, but they differ in how they exploit this relationship [15].

While this derivation was done using AWGN, the same method still applies in the presence of colored noise. The proof of eigenvalue/eigenvector relationships between $\mathbf{R_{xx}}$, $\mathbf{AR_{ss}A}^H$, and $\mathbf{R_{ww}}$ in the colored nose case can be shown via the theory of generalized eigenvectors [3], and if the noise covariance matrix is known, it can be used to whiten the receiver data prior to processing [15] and can be used in estimation of the incident signals [2].

2.4.1 Subspace Position Estimation

As discussed in Section 2.3, TDOA and FDOA require an iterative geolocation method such as Newton-Raphson because isochrones and isodops are curved contours. The subspace methods such as MUSIC are more related to basic interferometry with the direction finding output being a DOA. Since a DOA is essentially a vector with unknown magnitude, a direct solution can be determined from a set of collections of the signal from the same emitter, without having to iterate.

With only two collections the position estimation is trivial since it is simply the crossing

point of two vectors. With three or more collections, the vectors are unlikely to cross at the exact same position, and a least-squares estimate must be made. For the more general three-dimensional case, the distance of a point from a line can be defined as [16]

$$h = \frac{\|\mathbf{d} \times (\mathbf{a} - \mathbf{c})\|}{\|\mathbf{d}\|}, \tag{2.15}$$

where $\mathbf{a}$ is a point on the line, $\mathbf{d}$ is a vector along the line, $\mathbf{c}$ is the point in question, $\| \ \|$ indicates the 2-norm, and $\times$ indicates the cross product. For this application, $\mathbf{a}$ is the collector location, $\mathbf{d}$ is the DOA vector, and $\mathbf{c}$ is the estimated emitter location. A least-squares solution for multiple lines can be obtained by solving for the value of $\mathbf{c}$ for which the derivative of the sum of squares is zero:

$$\sum_{i=1}^{n} \frac{\mathrm{d}\left(h_i^2\right)}{\mathrm{d}\mathbf{c}} = \mathbf{0}, \tag{2.16}$$

where n is the total number of collections being used to estimate the position, the notation of a derivative with respect to a vector indicates that the derivative should be taken with respect to each dimension separately, and $\mathbf{0}$ is the zero vector. For the two-dimensional case, and additionally assuming $\mathbf{d}$ is a unit vector, expanding (2.15) results in

$$h = c_x d_y - c_y d_x - a_x d_y + a_y d_x. \tag{2.17}$$

By squaring (2.17), taking the derivative, and then summing the results for all n collections as specified by (2.16), we can convert the problem into an $\mathbf{AX} = \mathbf{B}$ type matrix equation where $\mathbf{c}$ can be solved for directly:

$$\sum_{i=1}^{n} \left(\begin{bmatrix} d_{yi}^2 & -d_{xi}d_{yi} \\ -d_{xi}d_{yi} & d_{xi}^2 \end{bmatrix} \right) \begin{bmatrix} c_x \\ c_y \end{bmatrix} = \sum_{i=1}^{n} \left(\begin{bmatrix} d_{yi}^2 & -d_{xi}d_{yi} \\ -d_{xi}d_{yi} & d_{xi}^2 \end{bmatrix} \begin{bmatrix} a_{xn} \\ a_{yn} \end{bmatrix} \right). \tag{2.18}$$

2.4.2 Signal Reconstruction

Once an estimate has been made of signal DOAs, the original signals can be reconstructed by using (2.8) and (2.1) to recreate the steering matrix and then solving for $\mathbf{s}$ in (2.6) as $\mathbf{s} = \mathbf{A}^{-1}\mathbf{x}$. Generally, the steering matrix will not be square, however, so its inverse cannot

be taken directly. Instead, the pseudo-inverse can be calculated. For an $m \times n$ matrix $\mathbf{A}$ where $m > n$ and the columns of $\mathbf{A}$ are linearly independent, the Moore-Penrose pseudo-inverse is defined as [17]

$$\mathbf{A}^{+} = \left(A^{H}A\right)^{-1}A^{H}. \tag{2.19}$$

Since $p > q$ and the steering matrix spans the signal subspace, its columns are linearly independent and this definition of the pseudo-inverse applies. Therefore, the original signals can be estimated as

$$\mathbf{s} = \mathbf{A}^{+}\mathbf{x}. \tag{2.20}$$

2.5 MUSIC

The MUSIC algorithm was proposed by Ralph Schmidt in 1979 [2] and was one of the earliest subspace methods. MUSIC develops from a recognition that since the eigenvectors of the array covariance matrix are a basis and are orthogonal to each other, the noise eigenvectors will be orthogonal to the signal eigenvectors. Since the steering vectors must by definition lie in the signal subspace, they will also be orthogonal to the noise eigenvectors. Recall from (2.8) that, given a known array center frequency and layout, the steering vector $\mathbf{a}$ is only a function of the signal DOA θ. The MUSIC algorithm is then a search over all θ in the visible range for the steering vectors which are most orthogonal to the noise eigenvectors of the array covariance matrix, resulting in peaks in the MUSIC DOA spectrum P [2]:

$$P_{MUSIC}(\theta) = \frac{\mathbf{a}^{H}(\theta)\mathbf{a}(\theta)}{\mathbf{a}^{H}(\theta)\mathbf{E_w}\mathbf{E_w}^{H}\mathbf{a}(\theta)}. \tag{2.21}$$

Note that the normalization in the numerator is only required when the steering vectors are not already normalized to magnitude 1. An example of the MUSIC spectrum using the parameters from the following section is plotted in Fig. 4. Note that the y-axis is labeled "Orthogonality"; since the denominator of the MUSIC algorithm is effectively an inner product, the MUSIC response at each DOA is an indication of how orthogonal the steering vector at each DOA is to the noise eigenvector(s). At exactly the correct DOA, assuming no receiver noise, the MUSIC response should be infinite because the inner product will be zero. Since the plots can have such a large dynamic range (theoretically infinite), the y-axis is plotted in decibels.

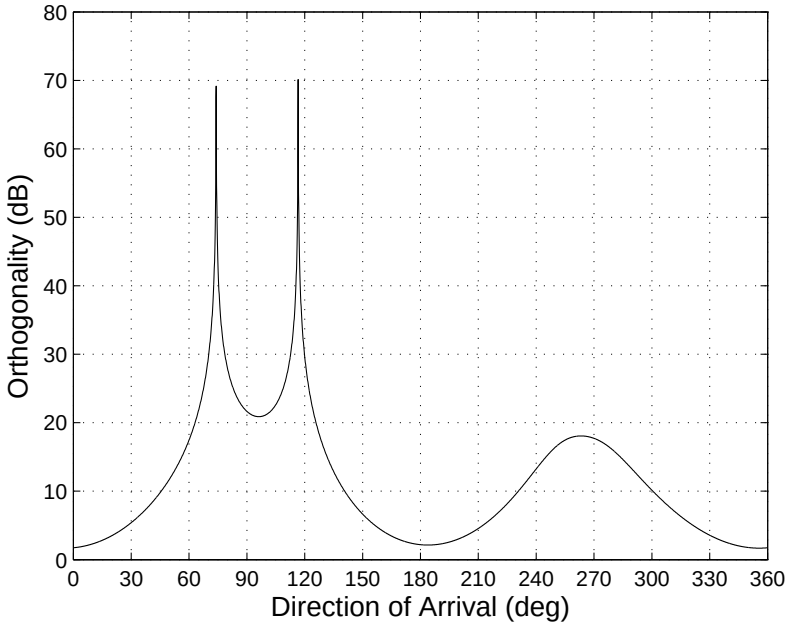


Figure 4. Sample MUSIC DOA spectrum for emitters at 74° and 116°.

In any real-world implementation with a finite number of samples with which to estimate the array covariance matrix, $\sigma_{\mathbf{w}}$ will not be exactly equal in all directions, and the resulting noise eigenvectors will not be exactly orthogonal to the original signal subspace. This will lead to slight errors in the MUSIC spectrum peaks for any particular collection period. MUSIC is asymptotically unbiased, however, so the spectra averaged over multiple collections will converge to the exact solution [2].

2.5.1 MUSIC Example

For the purposes of this example, let the receive elements be isotropic and arranged as the points on an equilateral triangle with receiver spacing of $\lambda/2$ and the origin at the center, as indicated in Fig. 5, where λ is based on the array center frequency of 200 MHz. Emitter 1 is transmitting a continuous cosine wave at 200 MHz and $\theta_1 = 116°$, and Emitter 2 is transmitting a 205 MHz cosine wave with $\theta_2 = 74°$, and we assume there is no additive noise in the receiver. The collection period is 100 ns and the sampling frequency is 800 MHz.

Since this array layout is not linear, the receiver array does not have a broadside, so

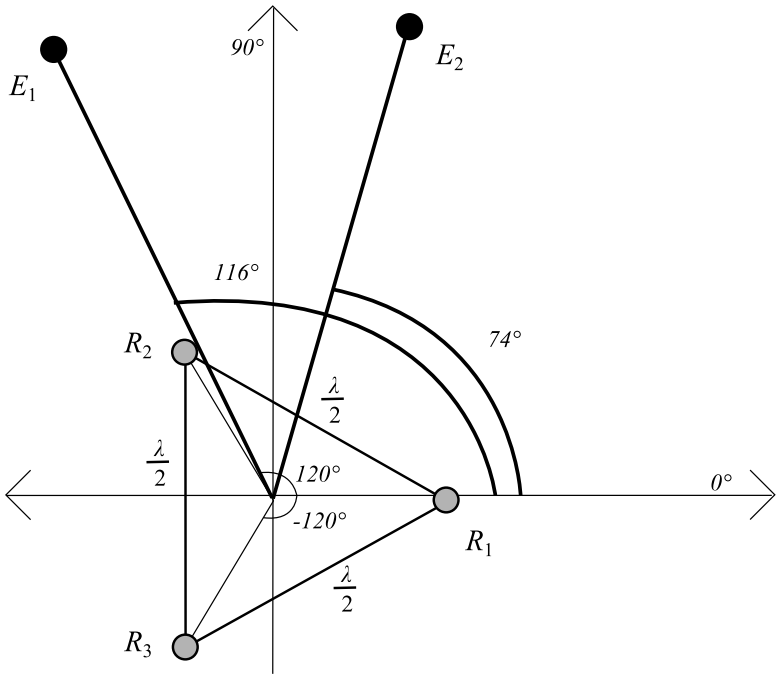


Figure 5. Example collection geometry.

the ϕ_{ij} terms in (2.8) must be defined slightly differently. For this arrangement let $\phi_{ij} = kd_j \cos\alpha_{ij}$, where d_j is the distance to the receiver from the origin and α_{ij} is the angle between the jth receiver, the origin, and the ith emitter, which is also equivalent to the difference between the angle to the emitter and the angle to the receiver relative to the x-axis. All three receivers are $\lambda/(2\sqrt{3})$ from the origin, R_1 is at 0° from the x-axis, R_2 is at 120° from the x-axis, and R_3 is at -120° from the x-axis, and the signal arrival angles relative to the x-axis have already been specified. The product

$$kd_j = \frac{2\pi}{\lambda}\frac{\lambda}{2\sqrt{3}} = \frac{\pi}{\sqrt{3}} \tag{2.22}$$

and the relative angles α_{ij} will equal

$$\alpha = \begin{bmatrix} 74-0^\circ & 116-0^\circ \\ 74-120^\circ & 116-120^\circ \\ 74+120^\circ & 116+120^\circ \end{bmatrix} = \begin{bmatrix} 74^\circ & 116^\circ \\ -46^\circ & -4^\circ \\ 194^\circ & 236^\circ \end{bmatrix}. \tag{2.23}$$

The steering matrix is then

$$\mathbf{A} = \begin{bmatrix} e^{j\frac{\pi}{\sqrt{3}}\cos 74^\circ} & e^{j\frac{\pi}{\sqrt{3}}\cos 116^\circ} \\ e^{j\frac{\pi}{\sqrt{3}}\cos -46^\circ} & e^{j\frac{\pi}{\sqrt{3}}\cos -4^\circ} \\ e^{j\frac{\pi}{\sqrt{3}}\cos 194^\circ} & e^{j\frac{\pi}{\sqrt{3}}\cos 236^\circ} \end{bmatrix} = \begin{bmatrix} e^{j0.500} & e^{-j0.795} \\ e^{j1.260} & e^{j1.809} \\ e^{-j1.760} & e^{-j1.014} \end{bmatrix}. \tag{2.24}$$

Since $\mathbf{w} = 0$ and $\mathbf{s} = \begin{bmatrix} e^{j(2\pi f_1 t)} & e^{j(2\pi f_2 t)} \end{bmatrix}^T$, where $f_1 = 200$ MHz and $f_2 = 205$ MHz, the received signal vector can now be defined:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{w} = \begin{bmatrix} e^{j(2\pi f_1 t + 0.500)} + e^{j2\pi f_2 t - 0.795)} \\ e^{j(2\pi f_1 t + 1.260)} + e^{j(2\pi f_2 t + 1.809)} \\ e^{j(2\pi f_1 t - 1.760)} + e^{j(2\pi f_2 t - 1.014)} \end{bmatrix}. \tag{2.25}$$

The sample estimated array covariance matrix $\mathbf{R_{XX}}$ must be determined numerically; using MATLAB to generate the 3×100 received signal matrix and 3×3 covariance matrix results in

$$\mathbf{R_{XX}} = \begin{bmatrix} 0.7753 & -0.0901 - j0.7374 & 0.2370 + j0.6537 \\ -0.0901 + j0.7374 & 2.6818 & -2.7092 + j0.6237 \\ 0.2370 - j0.6537 & -2.7092 - j0.6237 & 2.8917 \end{bmatrix} \tag{2.26}$$

which, after an eigen-decomposition, has the eigenvalues and eigenvectors which are listed in Table 1. Note that the third eigenvalue is zero because there is no signal (or noise) present in this dimension; therefore, the third eigenvector is the only noise eigenvector.

Table 1. Circular array eigenvectors and eigenvalues.

eigenvalue	5.7753	0.5734	0
eigenvector	$0.0679 + j0.1873$	$-0.3326 - j0.9175$	$0.0304 + j0.0838$
	$-0.6637 + j0.1528$	$-0.0706 + j0.0163$	$0.7101 - j0.1635$
	0.7046	0.2059	0.6791

The MUSIC DOA spectrum can then be obtained using (2.21); the result has already been presented in Fig. 4, and the peaks are at the source signal DOAs of 74° and 116°. The additional hump at 260° is a near orthogonality which occurs at the mirrored DOA angles. Because the plot peaks approach infinity but will not typically equal it due to

additive noise, limitations in the angular step size which is searched, and rounding errors during processing, the DOA spectrum is typically plotted in decibels so that the various peak heights are visually similar.

The sharp-eyed reader may have noticed a discrepancy in the above example, in that the wavelength used to construct the steering vector matrix in (2.24) is the array center wavelength of 200 MHz for both signals, despite the fact that one of the signals is not at the center frequency. This arises because the exact source signal frequencies are not known a priori to conducting the subspace analysis to separate the signals, and one must assume the array center frequency. The resulting DOA determined for the non-center frequency signal will have an induced error which is proportional to the frequency offset. Methods of correcting for this offset and for the effects of wideband signals are the focus of the rest of this thesis.

2.5.2 MUSIC Visualization

As alluded to in the eigen-space discussion, the relationship between $\mathbf{x}$, $\mathbf{A}$, and $\mathbf{s}$ can also be viewed geometrically. If the set of all possible steering vectors $(\theta = 0 : 360^\circ)$ is determined and plotted in p-space, it will transcribe a continuous q-dimensional object. For $p = 3$ and $q = 2$ this will be a line in three dimensional space; for $p = 4$ and $q = 3$ this will be a surface in four dimensional space. If the array is linearly arranged, the steering vector matrix will transcribe two overlapping 180° arcs; if it is not linearly arranged (i.e. two- or three-dimensional), it will transcribe a closed loop.

If the signals are adjusted to be real vice analytic, the example above can be plotted in three-dimensional space, where each dimension represents one of the receiver elements. By doing this, the signal and noise eigenvectors can be visually distinguished, and the orthogonality of the steering vectors in the signal DOAs to the noise eigenvector can also be visually ascertained.

The results are plotted in Figs. 6 and 7. Even though the source signals are real, since the steering vectors are complex phase terms, the received signals are complex, as are the covariance matrix and eigenvectors, and the problem has to be separated into real and imaginary portions to be depicted . In both of these figures, the gray portions are the received signals $\mathbf{x}$, and the blue lines are the eigenvectors of $\mathbf{R_{XX}}$. From visual inspection, two of

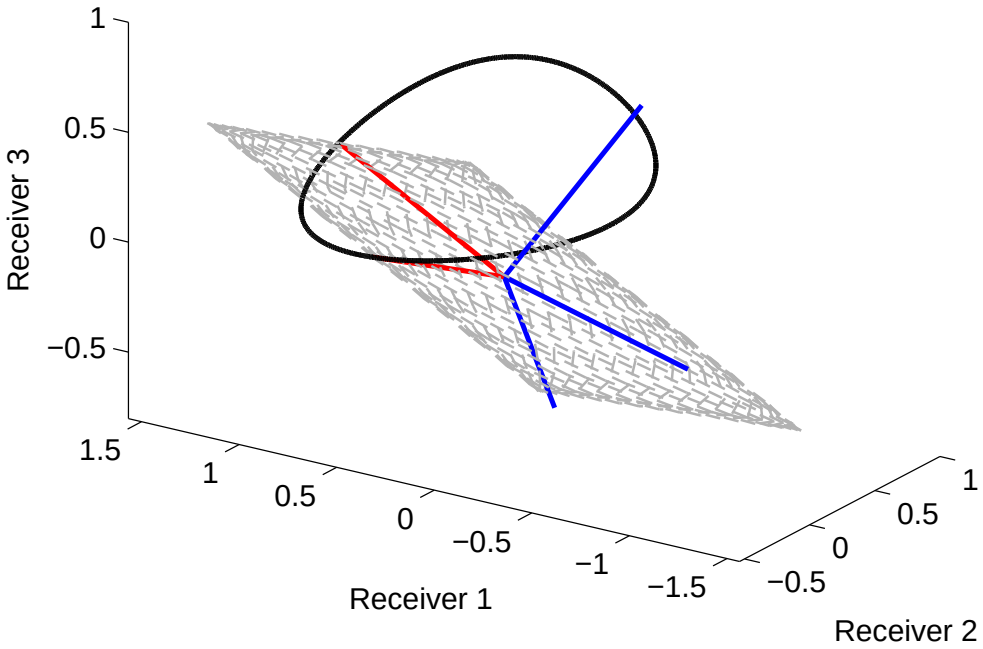


Figure 6. MUSIC geometric visualization – real portion.

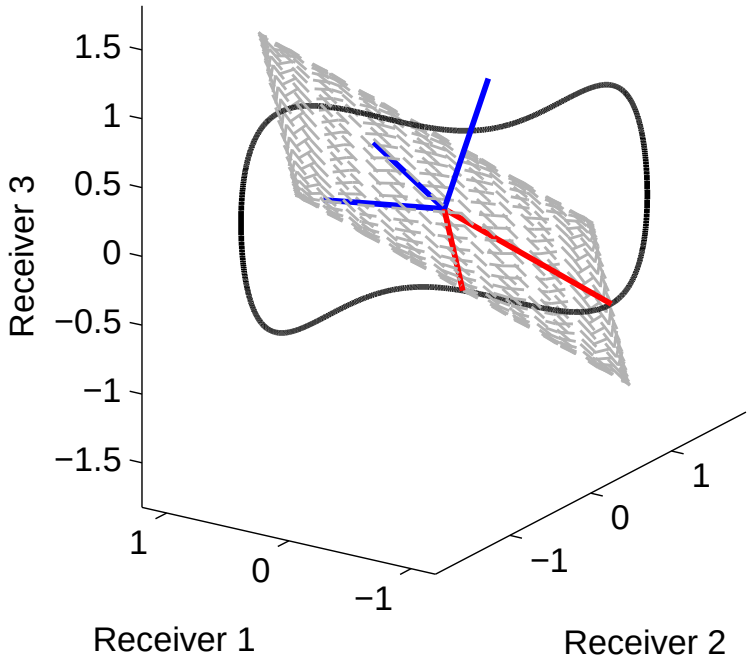


Figure 7. MUSIC geometric visualization – imaginary portion.

the eigenvectors in both the real and imaginary portion plots lie in the signal subspace and the third is normal to the signal subspace; this third eigenvector is the noise eigenvector.

The black loop in both figures is the continuum of all steering vectors, and the red lines indicate the steering vectors $\mathbf{a}(\theta_1)$ and $\mathbf{a}(\theta_2)$ at the signal DOAs. By visual inspection the red steering vectors lie in the gray signal subspace and are orthogonal to the noise eigenvector.

What the relationship looks like for the real portion of the signals when noise is included at a 10 dB SNR is shown in Fig. 8. Because of the additive noise, the received signal subspace is no longer only a two-dimensional plane. However, there are two dominant dimensions which equate to the two largest eigenvalues, and the subspace dimension with the smallest eigenvalue is still the noise subspace. The orthogonal relationship of the steering vectors to the noise eigenvector remains despite the noise.

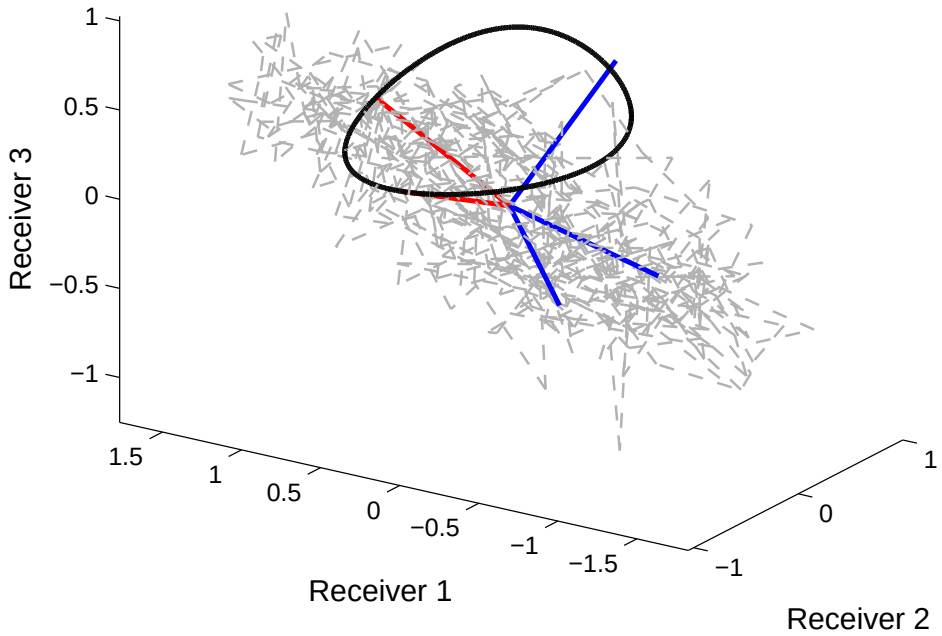


Figure 8. MUSIC geometric visualization – real portion including noise.

2.6 Emitter Location Processing Review

In this chapter, the principles behind various direction finding and geolocation methods were presented, with particular emphasis on explaining the derivation and providing an example of the MUSIC subspace algorithm. In the next chapter, the MUSIC algorithm is investigated to determine its performance characteristics for a sparse receiver array in the presence of signals with varying frequency, bandwidth, DOA, and SNR.

CHAPTER 3:
MUSIC Performance Analysis

3.1 Analysis Setup

For the purposes of this analysis, a linear array of six receivers on the x-axis with a visual range of -90° to 90° relative to array boresight is modeled. Additionally, it is assumed that all of the receive elements are isotropic. The array center frequency f_c is 1 GHz, the search bandwidth is 500 MHz to 1500 MHz, and sampling is conducted at 4 GHz. It should be noted that this analysis is conducted at radio frequency (RF); see Appendix Section A.3 for a discussion of implementing MUSIC at intermediate frequency (IF). This receiver array is also referred to as the collector.

3.1.1 Array Configuration

In order to ensure low sidelobes and no ambiguities, the ideal receiver spacing would be a uniform spacing of one half wavelength at the desired collection center frequency. With six receive elements, such an array has a very short baseline of only two and a half wavelengths and a relatively wide beamwidth. The direction finding resolution of an array varies with the processing method and the signal parameters, but in general it is proportional to the beamwidth and inversely proportional to the baseline; therefore, a large baseline is usually desirable. With uniformly spaced elements, however, a large baseline means that many receivers are required, which increases the weight and design complexity. Additionally, receive element size may prevent the placement of elements as close as $\lambda/2$. Airborne and spaceborne applications tend to be weight and space limited, so for those applications a sparse layout may be more desirable. The tradeoff with a sparse array is that ambiguities may exist and the sidelobes are higher than with a uniformly spaced array. These tradeoffs can be mitigated somewhat through careful design, such as by accepting ambiguities outside of a particular limited visible range.

The sparse layout used throughout the rest of this chapter is

$$d = \begin{bmatrix} 0 & 0.5 & 1.5 & 3.5 & 6 & 10 \end{bmatrix} \text{ wavelengths}, \tag{3.1}$$

where d_i represents the location of a receiver relative to the origin; defining the first receiver at the origin serves to help simplify the calculations. For the same number of elements, this array has a four times increase in baseline over the uniformly spaced array and a narrower main beamwidth. The sidelobes for this sparse array are significantly higher than for the uniformly spaced array, however, as indicated in Fig. 9. In fact with this spacing scheme, there are no ambiguous lobes, but all sidelobes are only $\sim$6 to 8 dB down from the main beam. A depiction of the resolution difference between the uniform and sparse arrays using the MUSIC algorithm is shown in Fig. 10. In this example there are two sinusoidal signals at a 2° separation relative to the collector, but the uniform array (Fig. 10(a)) cannot resolve them both and its MUSIC spectrum only has one peak, while the MUSIC spectrum of the sparse array (Fig. 10(b)) clearly has two peaks. The reason is that the larger baseline of the sparse array results in a narrower MUSIC response. Additionally, it is important to point out that what appears as noise in Fig. 10(b) is actually the response of the array's sidelines to the signals present.

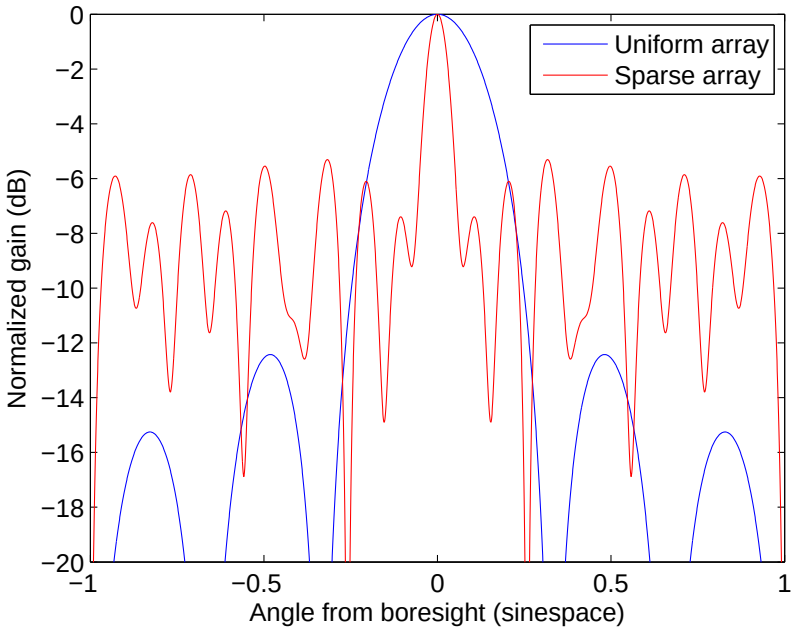
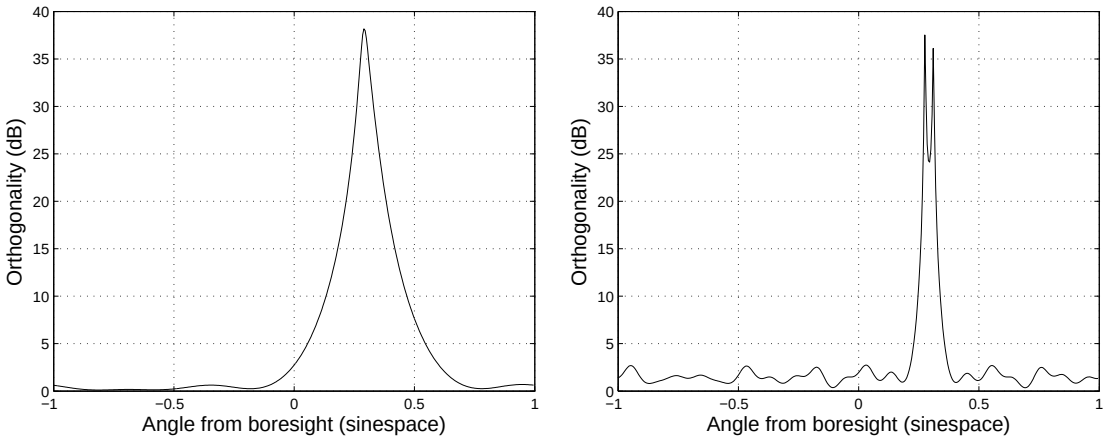


Figure 9. Array patterns with six isotropic elements.

At this point it should also be noted that in general, results are plotted in sinespace vice degrees, where sinespace is defined as the sine of the angles in the visible range. This is



(a) Uniform array, $\lambda/2$ spacing. (b) Sparse array of (3.1)

Figure 10. Six-element array resolution for 2° emitter separation.

because the MUSIC response of a signal is the same width across all of sinespace, whereas the MUSIC responses plotted in degrees widen out as the signals are further from boresight. Plotting in sinespace ensures that the primary distortions will be from signal bandwidth and array sparsity. A sinespace step size of 0.001 is used for all of the scenarios in this chapter.

3.1.2 Emitter Parameters

The incident signals being modeled are variations of three basic signal types: pure sinusoid, linear frequency modulation (LFM), and quadrature phase-shift keying (QPSK). All LFM signals are modeled as an "up-chirp" centered on a desired frequency, and all QPSK signals utilize rectangular pulse shapes with phase terms generated using the `randi` function in MATLAB. It should be noted at this point that pulsed QPSK modulation is not utilized in phase-coded radar systems. This signal type has been chosen for this thesis because it has a bandwidth profile which provides a reasonable approximation to a bi-phase or polyphase coded radar pulse as well as a short segment of a phase modulated communications signal. Radar terminology for a phase-coded signal will be used when describing the QPSK signals: a code is the full set of phases transmitted during the QPSK pulse, and a subcode is an individual phase term transmitted during one subpulse [18]; in communications terminology the subcode would be a channel symbol [19].

Emitters are positioned across the visible range at distances between 100 and 200 nmi,

and are assumed to be stationary. All of the emitted signals are modeled as pulses which arrive at the array at time zero and last for 100 ns. The emitter parameters being varied in this analysis are frequency, bandwidth, DOA, and SNR. For the purposes of simplifying signal creation and processing, all signals are generated as analytic signals in the time domain.

3.1.3 Collector Motion

When multiple collections are simulated, the collector is modeled as being mounted on the side of an aircraft which is moving in the positive x direction at 300 knots. The altitude is not being modeled as this analysis is conducted in only two dimensions; therefore, the emitters are assumed to be in the same plane as the collector. For an aircraft at altitude this could be the slant plane, though the intersection of that plane with the earth is not being utilized as a constraint in the position estimation which was delineated in Section 2.4.1.

Collections are made every two minutes, and all signals being modeled are assumed to be received at every collection time. Doppler shift due to collector motion is not included in the model, and the collector is assumed to be stationary for the duration of a collection. This first assumption simplifies the association of signals over multiple collections, and the latter two assumptions are reasonable based on the short collection durations.

3.1.4 Processing Method and Results Presentation

For this analysis the frequency domain form of (2.6) was utilized (see Appendix A.2). Working in the frequency domain provides some advantages because the signal frequencies are more readily accessible without additional processing and band-limiting can be utilized to reduce computation and noise influence. The source signals are generated in the time domain, and the discrete Fourier transform (DFT) (specifically MATLAB's implementation of the fast Fourier transform (FFT)) is then used to convert the received signals to the frequency domain for all further processing.

An overhead "map view" of the collection plane may also be included with the results, especially when emitter motion is modeled. In these views the collector position(s) are indicated by a black circle and the actual emitter positions by a black cross. Estimated DOA lines colored to correspond to each received signal are plotted extending from the

collector positions and, when applicable, black diamonds are used to indicate the least squares estimated emitter positions.

3.2 Frequency Effects

In order to gain an understanding of the effects of frequency on the subspace DOA spectrum, three sinusoidal signals at the frequencies and locations in Table 2 were modeled over five collections. No receiver noise was included, ensuring that only the frequency variations would affect the results.

Table 2. Signals utilized in modeling frequency effects.

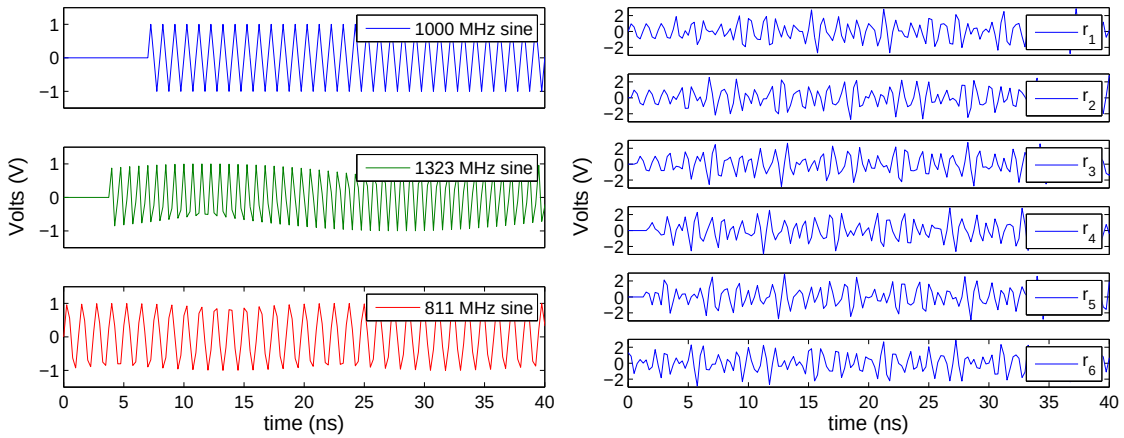
Signal Number	Frequency (MHz)	location (nmi)
1	1000	[-50, 90]
2	1323	[-40, 190]
3	811	[-100, 110]

The results are presented in Fig. 11. The MUSIC results for all five collections are overlaid in Fig. 11(c) so that the relative change in emitter locations can be seen. From Fig. 11(d), it can be seen that the DOA lines only intersect in the close vicinity of one of the three actual emitter locations - Emitter #1, the one at the array center frequency.

As initially mentioned in Section 2.5.1, this direction finding and geolocation error arises because the center frequency was assumed when constructing the steering vectors for use in (2.21). There exists, however, a relatively simple solution which requires understanding of the origin of the steering vectors. Without loss of generality, the solution can be found by examining the noiseless relationship of the ith emitter with the jth collector from (2.8):

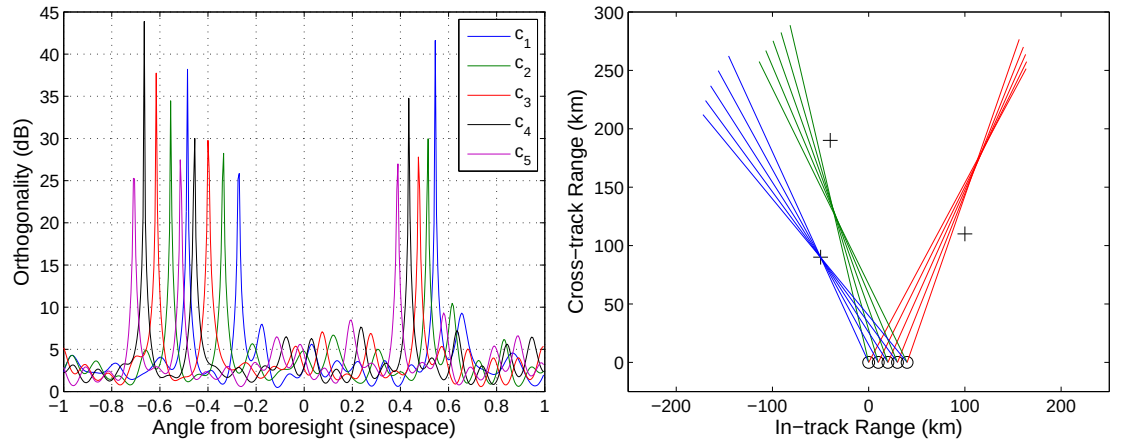
$$x_{ij} = s_i e^{j2\pi \frac{d_j}{\lambda_i} \sin\theta_i}. \tag{3.2}$$

The effect of the exponential term is simply a phase shift, and the received signal magnitude s_i will be the same regardless of the DOA. For a particular collection, x_{ij}, s_i, and the magnitude of the phase shift ϕ_{ij} are fixed. Note that there are essentially two unknown variables in the phase shift, however: the wavelength λ_i and the DOA θ_i, which means that there is a dependent relationship between the wavelength, the DOA, and the phase shift and that there are an infinite number of wavelength/DOA pairs which taken together provide the correct phase shift. In order to solve for the DOA using MUSIC, an assumption was made



(a) Transmitted signals; first 40 ns.

(b) Received signals; first 40 ns.



(c) MUSIC DOA spectrum.

(d) Map view of DOAs.

Figure 11. Results of MUSIC processing on narrowband signals with different frequencies.

about the wavelength; therefore, if the wavelength is incorrect then so is the resulting DOA. The value of the phase shift is fixed, so an equality can be configured to solve for the actual DOA as

$$\phi_{ij} = 2\pi \frac{d_j}{\lambda_c} \sin\theta_{i,MUSIC} = 2\pi \frac{d_j}{\lambda_{i,est}} \sin\theta_{i,est}, \tag{3.3}$$

which can be further reduced to

$$f_c \sin\theta_{i,MUSIC} = f_{i,est} \sin\theta_{i,est}, \tag{3.4}$$

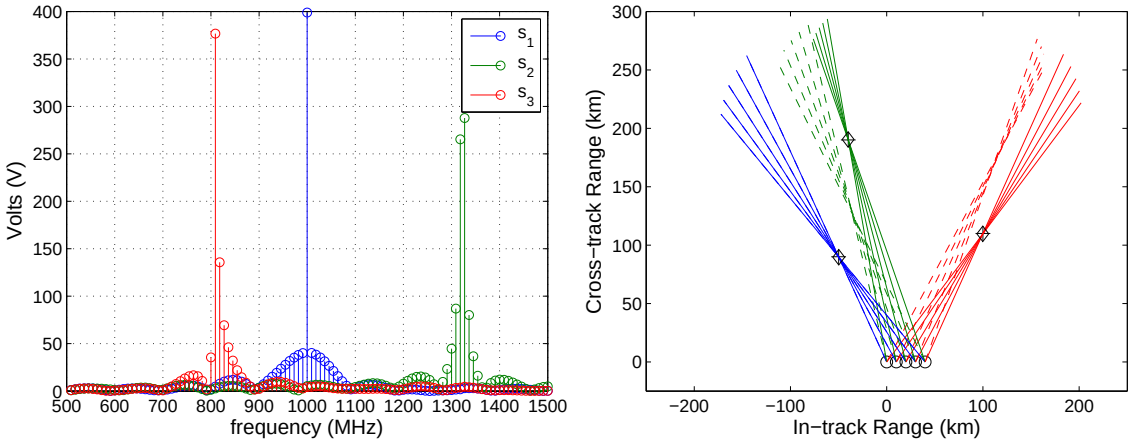
where λ_c is the wavelength at the array's center frequency at which the MUSIC algorithm was run, $\lambda_{i,est}$ and $f_{i,est}$ are the ith signal's estimated wavelength and frequency, and $\theta_{i,MUSIC}$ is the DOA of the ith signal as determined by the MUSIC algorithm. There are still two unknowns: $f_{i,est}$ and $\theta_{i,est}$. Since the signal parameters of s_i can be determined via signal reconstruction, (Section 2.4.2), the center frequency of the signal can be estimated and the true DOA can be estimated as

$$\theta_{i,est} = \arcsin\left(\frac{f_c}{f_{i,est}} \sin\theta_{i,MUSIC}\right). \tag{3.5}$$

Attempting to accurately reconstruct the signal with an incorrect DOA may seem bound to result in errors; however, it can easily be proved to provide an accurate result. Recall that signal reconstruction involves an estimation of the steering vectors for each signal, and that the steering vectors are simply the collection of phase shifts applied to each signal at each receiver. As stated previously, the magnitude of the phase shifts are already known, at least to the accuracy that the DOAs can be extracted from the MUSIC DOA spectrum.

It is important to note that the signal frequency can only be estimated from the reconstructed signal due both to the nature of the DFT and the presence of noise in an actual collection. A method for estimated the average frequency in a signal's discrete frequency response follows. First, normalize the magnitude of the reconstructed signal's frequency spectrum and filter to the desired search bandwidth. Next, reduce the impact of noise and the signal not being centered in the search bandwidth by setting the lowest 10% of the frequency spectrum values to zero. Finally, a weighted mean frequency can be determined by taking the inner product of the vector of DFT frequency bins with the normalized and truncated frequency spectrum and then dividing by the sum of the normalized and truncated frequency spectrum. The reconstructed frequency spectrum for the first collection is provided in Fig. 12(a), and estimated frequencies for each of the five collections are presented in Table 3. By comparison with the original frequencies listed in Table 2 it can be seen that this method works quite well for these narrowband signals.

Using (3.5) with the values in Table 3, we can calculate frequency-corrected DOAs for each collection. A map view with the original DOAs as dashed lines and the corrected DOAs as solid lines is illustrated in Fig. 12(b). Additionally, the estimated emitter positions



(a) Reconstructed signal frequency spectrum. (b) Map view with frequency-corrected DOAs and estimated emitter locations.

Figure 12. MUSIC DOA frequency correction.

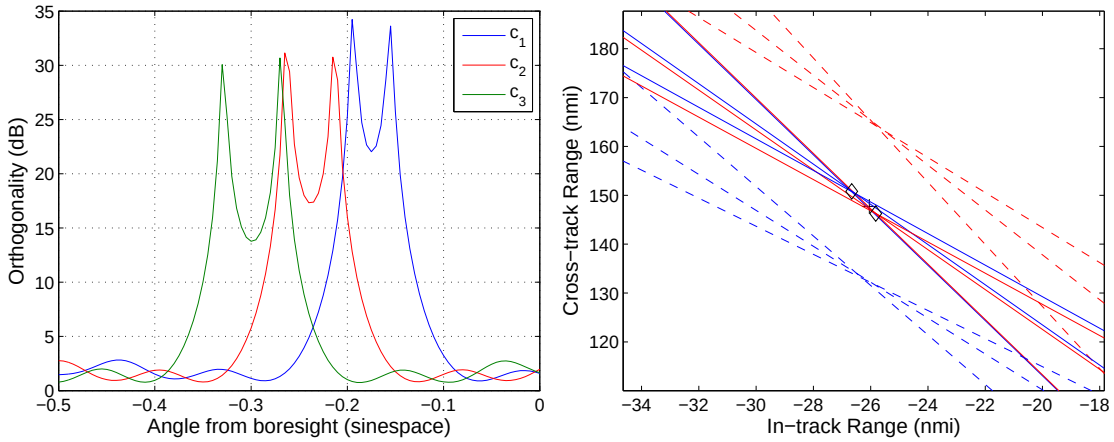
Table 3. Estimated signal frequencies.

Collection	f_1 (MHz)	f_2 (MHz)	f_3 (MHz)
1	1000.0	1323.2	810.1
2	1000.0	1323.0	810.1
3	1000.0	1322.2	810.2
4	1000.0	1322.2	810.2
5	1000.0	1323.2	810.0

have been plotted (diamonds), and one can see that they align very well with the actual positions (crosses).

An interesting artifact of this frequency effect is that MUSIC can easily resolve two signals which emanate from the exact same location as long as they are different enough in frequency. The MUSIC DOA spectra for two QPSK signals at frequencies of 900 and 1100 MHz which are co-located at -10° from collector boresight are illustrated in Fig. 13(a). Both signals also have a bandwidth of 20% f_c and a 10 dB SNR. There are clearly two distinct peaks in the MUSIC spectra for each collection, but from the map view (Fig. 13(b)) it can be seen that the corrected DOAs overlap with estimated locations very close to the correct position. It should be noted that this effect also depends on the angle from array boresight; if the signals are at boresight, there will be zero relative phase shifts between the elements, and the frequency difference will have no effect and there will be a

single MUSIC peak.



(a) MUSIC DOA spectra. (b) Map view.

Figure 13. Two 10 dB SNR QPSK signals with center frequencies of 900 MHz and 1100 MHz and the same source location.

This frequency/DOA dependence can also have the alternate effect of causing two signals with otherwise sufficient angular separation to overlap because of their differences in frequency, making them unresolvable in that collection.

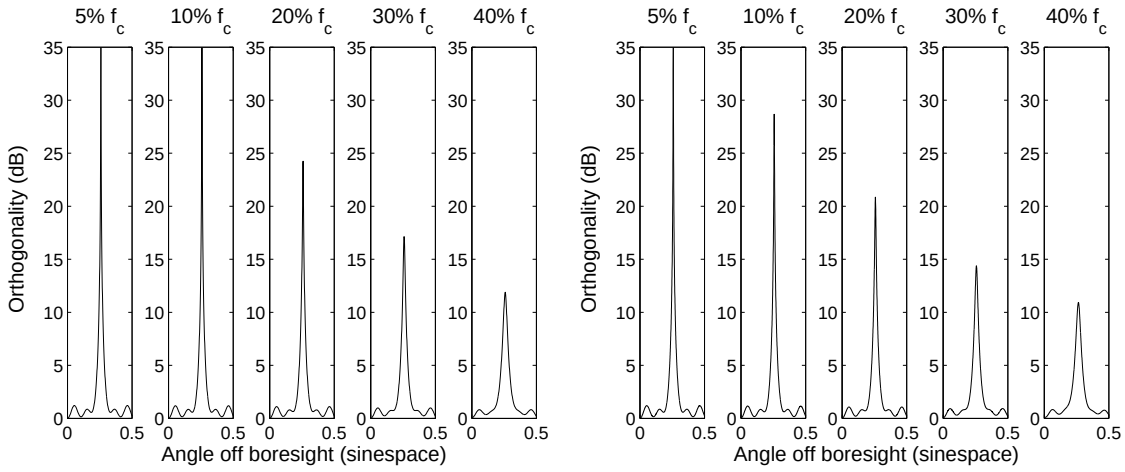
3.3 Bandwidth Effects

For the initial bandwidth analysis, all signals were centered at the array center frequency of 1 GHz, and bandwidths of 50 MHz, 100 MHz, 200 MHz, 300 MHz, and 400 MHz were tested, which are respectively 5%, 10%, 20%, 30%, and 40% of the center frequency f_c. Additionally, all signals were set at a DOA of 15° with no receiver noise. Only the LFM and QPSK signal types were implemented since the bandwidth of the sinusoidal pulse is fixed at $B_p = 1/t_p$, where t_p is the pulse duration. For this analysis, $t_p = 100$ ns and $B_p = 10$ MHz, which is approximately equal to the width of one FFT bin.

The bandwidth of an LFM signal is being defined as $B_{LFM} = f_{hi} - f_{lo}$, where f_{hi} is the highest instantaneous frequency transmitted and f_{lo} is the lowest instantaneous frequency transmitted. The bandwidth of a phase coded signal such as QPSK is being defined as $B_{QPSK} = 1/T_s$, where T_s is the subcode duration [19]. Both of these are noise-equivalent bandwidth definitions assuming an infinitely long pulse; the true bandwidths as viewed on

a spectrum analyzer would be slightly larger due to the pulsed nature of the signals being modeled. For a pulse duration of 100 ns, however, this additional bandwidth is a negligible fraction of the carrier frequency as noted in the previous paragraph.

The MUSIC DOA spectra for a single LFM and QPSK signal at different bandwidths are presented in Fig. 14. The change in MUSIC response for both signal types appears very similar with a relatively linear reduction in peak amplitude inversely proportional to the change in bandwidth.



(a) LFM signal. (b) QPSK signal.

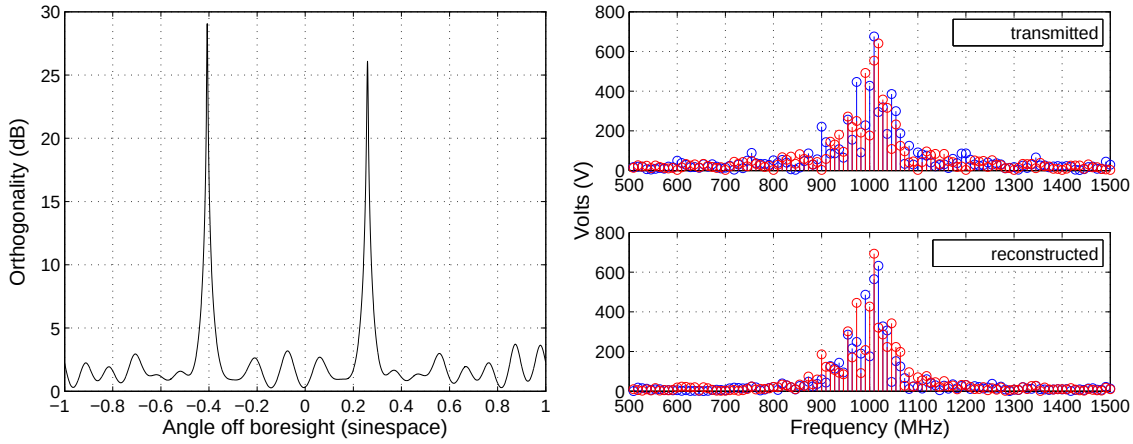
Figure 14. Effect of bandwidth on MUSIC spectrum.

This is likely because even though the bandwidths are the same, a phase coded signal has a sinc-shaped frequency spectrum where most of the energy is concentrated near the center frequency, whereas the energy in a LFM signal is evenly spread across the bandwidth since the frequency spectrum is approximately rectangular.

The effect of bandwidth still appears relatively minor, however, since a peak still remains at the proper DOA even if it is reduced in amplitude. Once multiple received signals are evaluated, signal bandwidth can have more noticeable effects on the MUSIC spectrum.

One might believe that signals with overlapping bandwidth would lead to problems with determining an accurate DOA, but that appears to not be the case in at least a basic two-signal scenario. The MUSIC DOA spectrum and the transmitted signal frequency spectrum for two QPSK signals at -24°/100 MHz bandwidth and 15°/150 MHz bandwidth, both at

the array center frequency and with 10 dB SNR, are illustrated in Fig. 15. Despite the complete bandwidth overlap, the signals are both easily resolvable and reconstruction is fairly good, especially close to the signal center frequency.

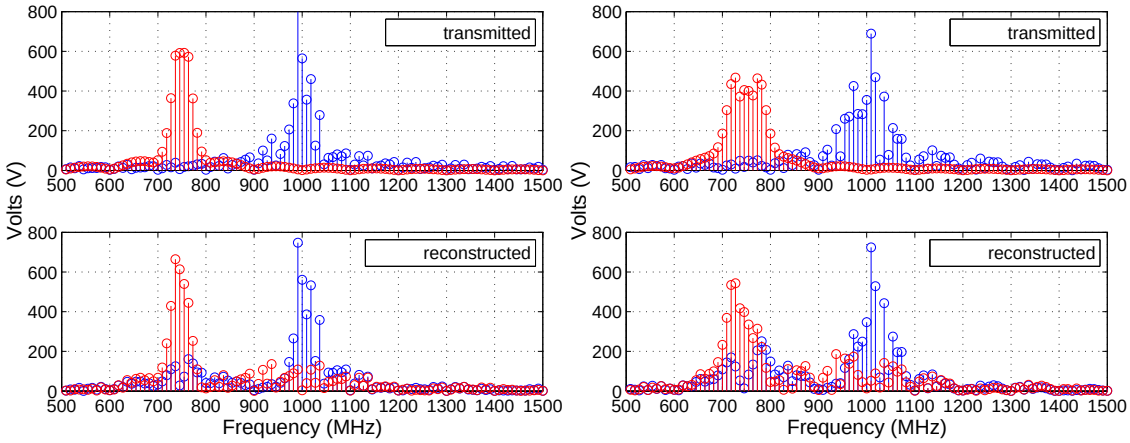


(a) MUSIC DOA spectrum. (b) Frequency spectra.

Figure 15. Effect of signal overlap on MUSIC DOA resolution and signal reconstruction.

It has already been shown that the sparse array being utilized can resolve narrowband signals with a small angular separation. If those signals are wideband, MUSIC can still resolve them, but other issues can arise even with a large frequency separation. Two examples utilizing a 750 MHz LFM at -30° and a 1000 MHz QPSK signal at -24° with no additive noise are provided in Fig. 16. Note that with the frequency adjustment in (3.5), the MUSIC algorithm "sees" the LFM signal at -22°, so this scenario has an effective signal separation of 2°. In Fig. 16(a) the signals have a bandwidth of 5% f_c, and in Fig. 16(b) the signal bandwidths are 10% f_c. From an examination of both figures, it can be seen that the signals have been blended in reconstruction, and the effect worsens as the relative bandwidth increases. This blending results from the various signal frequency bins being DOA shifted different amounts relative to the signal center frequency; since the signals are so close in angular separation, there is a crossing of frequency bins between signals which cannot be perfectly untangled using (2.20).

If a third signal with particular parameters is added to the previous 10% bandwidth scenario, another distortion occurs. One might expect that adding a signal with a large angular separation would have no impact and would be easily resolvable, but that is not the



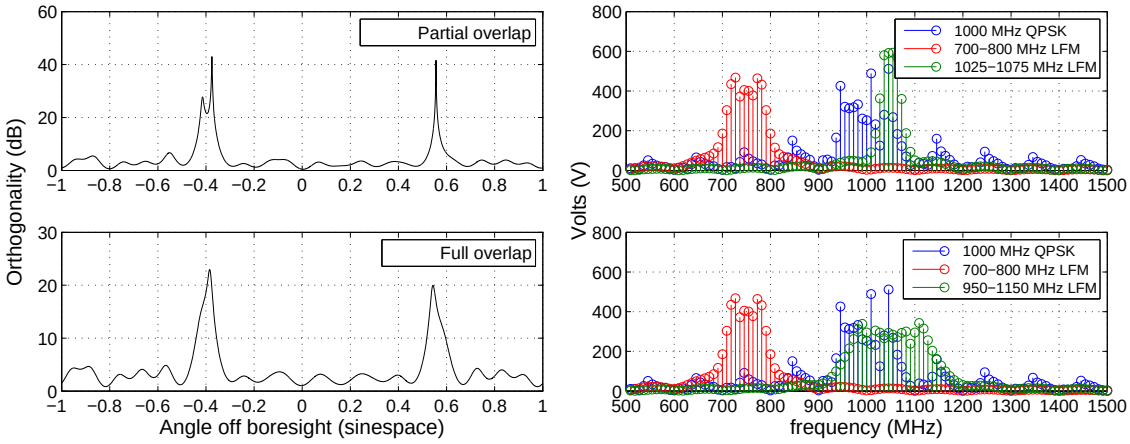
(a) MUSIC DOA spectrum. (b) Frequency spectra.

Figure 16. Transmitted and reconstructed frequency spectra of wideband signals with 2° angular separation.

case if the new signal overlaps in frequency with one of the prior two. For this scenario a 1030 MHz LFM signal at 32° has been added, with varying bandwidth of 50 and 200 MHz (5% and 20% f_c). The results are presented in Table 4 and Fig. 17. When the bandwidth of the third signal is increased to the point where it completely overlaps the QPSK signal, the MUSIC response becomes distorted. The two closely spaced signals are now no longer resolvable as separate signals. Additionally, the MUSIC response to the third signal is no longer the sharp peak it was with the partial overlap in bandwidth. From Table 4 it can be seen that the DOA estimations from the MUSIC spectrum in the fully overlapped case have been shifted from their closer-to-true estimations in the partially overlapped case.

Table 4. Effect of signal bandwidth on DOA with multiple closely spaced and frequency overlapped signals.

Signal	1000 MHz QPSK	750 MHz LFM	1030 MHz LFM
True DOA (frequency adjusted)	−24.0°	−22.0°	33.8°
MUSIC DOA with partial overlap	−24.5°	−22.1°	33.8°
MUSIC DOA with full overlap	−22.6°		32.9°



(a) MUSIC DOA spectra. (b) Frequency spectra of transmitted signals.

Figure 17. Effect of signal bandwidth on closely spaced and frequency-overlapped signals.

3.4 DOA Effects

It has already been noted that resolution decreases with increased DOA because the phase difference ϕ is a function of the sine of the DOA; as the DOA approaches 90° the DOA is changing much faster than the phase. For example, a sinespace step size of 0.01 at array broadside represents an angular step of 0.57°, while at 90° off boresight that same sinespace step represents an angular step of 8.11°. The DOA can have other affects on the MUSIC response as well. For this analysis, signals were generated at 1, 15, 30, 45, and 60° with no noise. Signals are centered at the array center frequency, and LFM and QPSK signals have a bandwidth of 20%. A depiction of the responses for a sinusoid is provided in Fig. 18. The peak amplitudes vary slightly, but that is only due to the discretization of the angular step impacting the exactness of the orthogonality response as certain steps are slightly closer to the actual DOA than others; recall that the ideal peak amplitude is infinite.

The MUSIC response for a LFM signal is provided in Fig. 19(a). For this signal type the response amplitude degrades fairly significantly, and the response widens as the angle from array boresight increases. The MUSIC response for a QPSK signal is illustrated in Fig. 19(b). Here, the QPSK MUSIC peak amplitudes degrade more slowly than those of a LFM signal, unlike in the bandwidth analysis above. This difference is likely due to the fact that a phase-coded signal has more energy concentrated near the array center frequency,

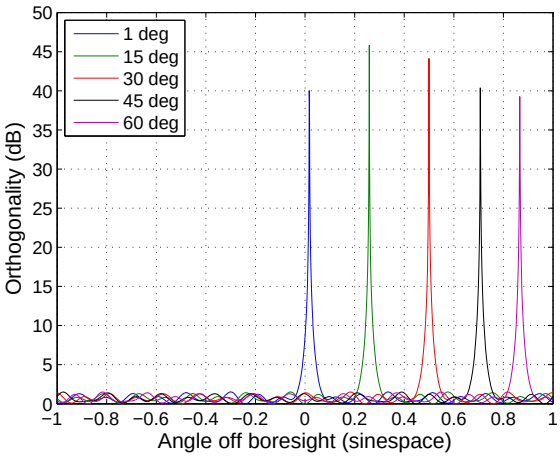
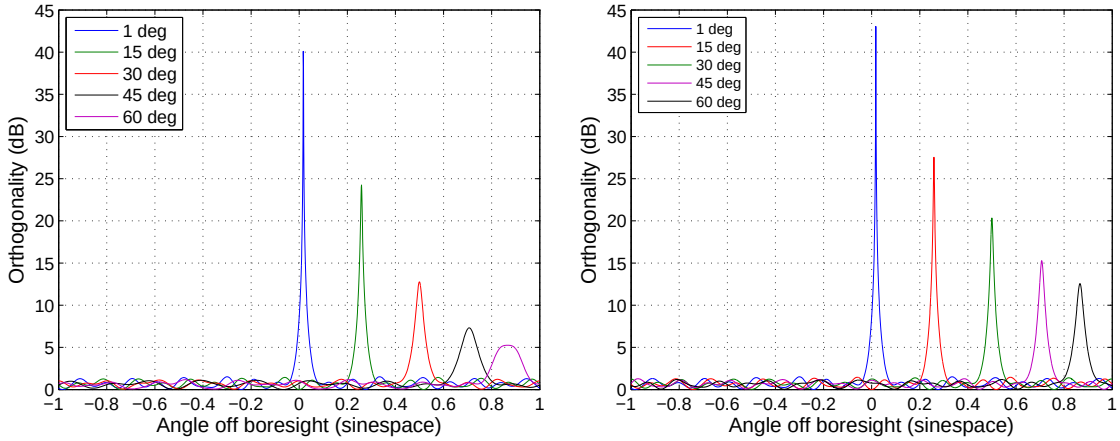


Figure 18. Effect of arrival angle on MUSIC spectrum with sinusoidal signal.

while a LFM signal has its energy spread more evenly over the entire bandwidth, so the additional distortion that occurs with increased DOA has less effect on phase-coded signals such as QPSK.



(a) LFM signal. (b) QPSK signal.

Figure 19. Effect of DOA on MUSIC spectrum with wideband signals.

That a degradation occurs at all is related to the frequency effects analysis of Section 3.2. As the DOA increases, the relative additional path distance δ between elements also increases, which results in an increased phase difference ϕ for frequencies not at the signal's center frequency. It is useful to think of each FFT frequency bin as affecting the MUSIC

response independently; bins above the signal center frequency map to higher DOAs and bins below the signal center frequency map to lower DOAs. The response spreads out over a larger set of DOAs and also decreases in amplitude because there is less energy in each bin as bandwidth increases.

3.5 SNR Effects

For this portion of the analysis, SNRs of 100, 10, 3, 0, -3, and -10 dB were utilized with Monte Carlo simulations of 100 trials. Signals are centered at the array center frequency with a DOA of 15°. The average MUSIC response at varying SNR for a sinusoidal signal with the six-element sparse array of (3.1) is illustrated in Fig. 20. A degradation of the peak amplitude with increased noise can be seen, similar to the effect due to bandwidth. In this case the likely reason is that since additive noise serves to rotate the signal eigenvectors of the covariance matrix away from their true locations, on average the steering vectors will be somewhat less orthogonal to the noise eigenvectors as the SNR decreases. Only the sinusoidal signal results are presented because the degradation rate is very similar for all signal types.

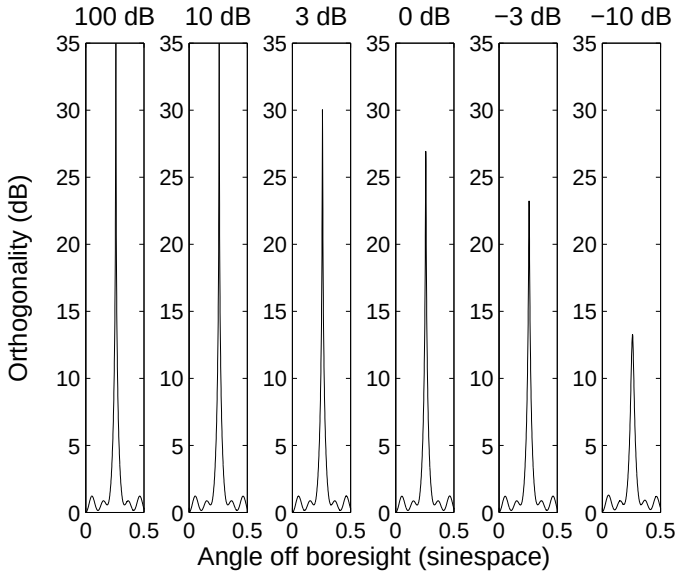


Figure 20. Effect of SNR on MUSIC spectrum with sinusoidal signal; average of Monte Carlo simulation with 100 trials.

The standard deviation σ_{DOA} and mean m_{DOA} of the estimated MUSIC peak location across the 100 trials are listed in Table 5; as the SNR decreases, the standard deviation increases, which should be expected. The mean DOA estimations remain very close to the true DOA of 15°, however, confirming that MUSIC is indeed an unbiased estimator. It should be noted that even the mean of the nearly noiseless case of 100 dB SNR is not exactly at the correct DOA of 15°; this is due to the finite sinespace step size of 0.001. In the vicinity of the correct DOA, the two closest angular values are 14.95° and 15.01°; thus, it can been seen that the algorithm converges on the closest angular step and not necessarily the exact DOA. This effect will add to the geolocation uncertainty but should average out over multiple collections at different DOAs.

Table 5. DOA statistics for varying SNR.

SNR (dB)	100	10	3	0	−3	−10
σ_{DOA}	1.07×10^{-14}	0.015	0.033	0.050	0.080	0.234
m_{DOA}	15.01	15.01	15.00	15.01	15.01	15.02

The role of the number of array receiver elements was also investigated to see if more receivers would result in an averaging effect which would improve the MUSIC peak amplitude for low power signals. A Monte Carlo simulation with 100 trials was run for a uniformly spaced array with 2, 3, 4, 5, and 6 elements and a sinusoidal signal at −10 dB SNR, with the averaged results presented in Fig. 21. The baseline was held constant at 2.5 wavelengths to ensure that changes in baseline did not affect the results. This does produce ambiguities at the lower receiver numbers, but the ambiguous results are outside of the 0–0.5 sinespace range which is plotted and can be ignored for this scenario. The relatively constant peak amplitude is somewhat surprising, as one might expect that by averaging across more receivers the effective SNR would improve.

The standard deviations and means at each array size are listed in Table 6. The means vary somewhat randomly, indicating that the estimated DOA accuracy is more a function of the baseline, which is constant in this scenario. The standard deviations steadily decrease with array size, indicating that the precision over multiple trials does improve with the number of receivers.

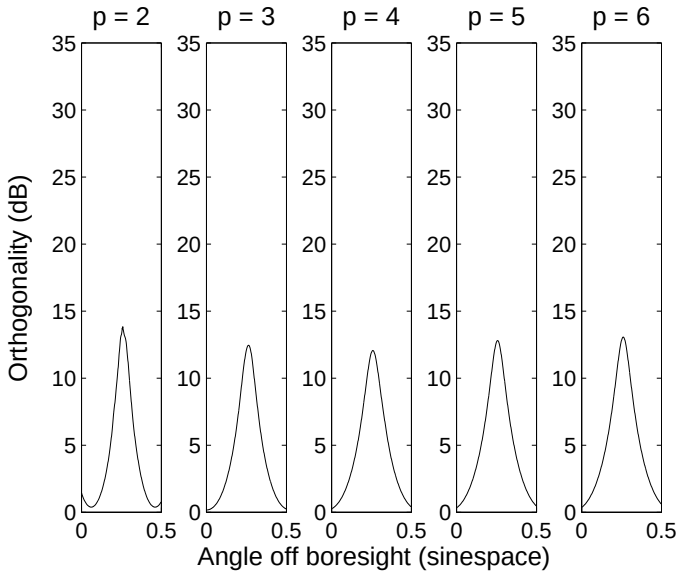


Figure 21. Effect of the number of array elements on the MUSIC spectrum with a sinusoidal signal at −10 dB SNR; average of Monte Carlo simulation with 100 trials.

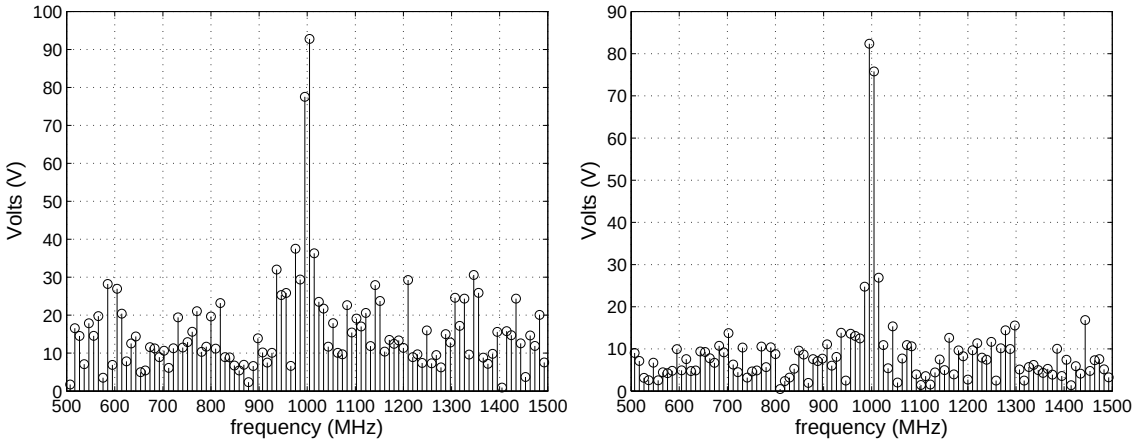
Table 6. DOA statistics for varying array size.

Array size	2	3	4	5	6
σ_{DOA} (deg)	1.69	1.31	1.22	1.02	0.85
m_{DOA} (deg)	15.12	15.15	15.01	14.88	15.12

While an averaging effect across multiple receivers did not affect the MUSIC peak amplitudes, it does impact the effective SNR of the reconstructed signals, as evidenced by Fig. 22, where the two-element reconstruction is visibly noisier than the six-element one.

Variations in SNR with multiple signals present causes additional distortions to the MUSIC DOA spectrum. The MUSIC spectra for two sinusoids of 1000 MHz and 1005 MHz at 15° and 18° with SNRs of 10, 3, and −10 dB are provided in Fig. 23. As the SNR decreases, the ability to resolve the peaks disappears. Signals further separated in frequency have the same result, so the effect is likely solely due to the SNR and not to any signal interaction in the frequency domain.

Next, the effect of signals with different SNRs was investigated. The MUSIC spectra for two sinusoids are presented in Fig. 24(a); the first is at 800 MHz and −5° and the second is at 1200 MHz and 15°. The SNR of the first is varied over 10, 0, and −10 dB, and the SNR



(a) Two-element array. (b) Six-element array.

Figure 22. Frequency spectrum of reconstructed 1 GHz sinusoid with received SNR of -10 dB.

of the second is held constant at 10 dB. The first peak nearly disappears when that signal is 20 dB less than the second. The large separation in DOA and frequency indicates that this is purely a SNR effect and is not caused by any frequency overlap.

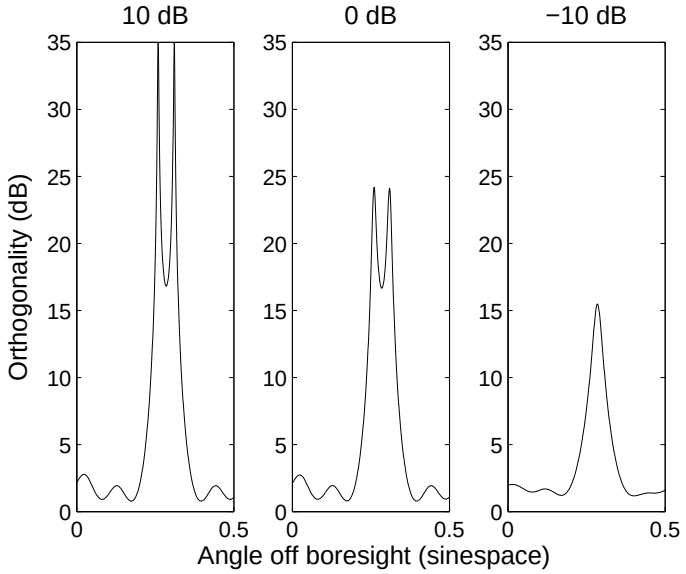
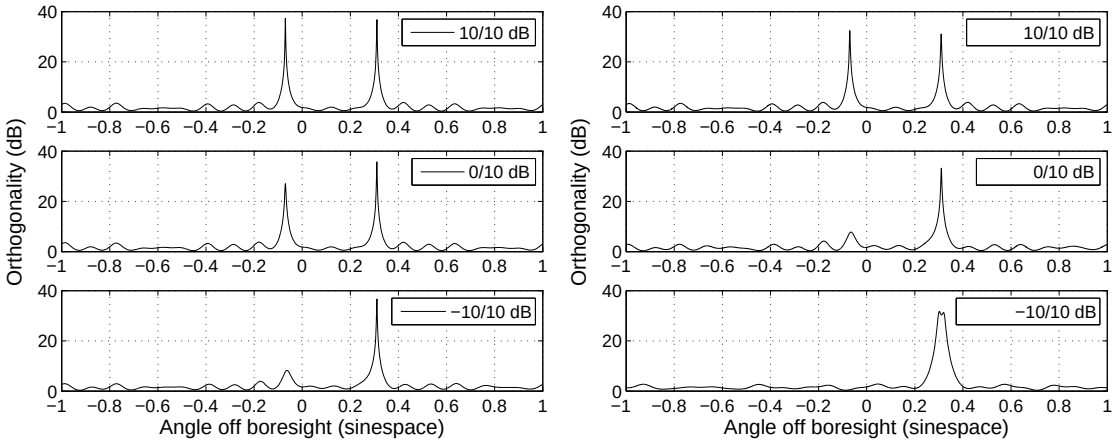


Figure 23. MUSIC spectrum effect of SNR on MUSIC spectrum DOA resolution with two sinusoids: 1000 MHz at 15° and 1005 MHz at 18°; average of Monte Carlo simulation with 100 trials.



(a) Sinusoidal signals. (b) 10% bandwidth QPSK signals.

Figure 24. MUSIC DOA spectra for signals with different SNRs.

The same trial was re-run with QPSK signals with a bandwidth of 10% f_c; the results are presented in Fig. 24(b). The peak amplitudes are reduced somewhat from the sinusoidal signals, as is expected based on results from Sections 3.3 and 3.4. In general, the drop in peak amplitude with increased SNR difference matches the results with sinusoidal signals. With wideband QPSK signals, there is an additional distortion of the MUSIC response of the high power signal. A possible reason for this distortion could be that since the array covariance matrix has very little information in the direction of the low-power signal, the low-power signal's subspaces becomes indistinguishable from the noise subspace. Since the MUSIC algorithm is being run under the assumption that two signals are present, this distortion of the subspace results in an incorrect estimate of the noise eigenvectors and a degraded MUSIC response.

3.6 Performance Analysis Review

An analysis of the MUSIC algorithm has been performed with particular emphasis on the effects of signal frequency, bandwidth, DOA, and SNR. In most scenarios, MUSIC performs very well even with wideband signals. The use of a receive antenna array with sparse spacing and -6 dB sidelobes likely contributes to some of the distortions and degradations in the MUSIC response, but the tradeoff in resolution over a uniformly spaced array may be worth the occasional inaccuracy, especially when averaged over multiple collections.

While this analysis was completed solely with the MUSIC algorithm, all of the subspace algorithms are based on leveraging the same array covariance matrix eigen-structure and signal/noise subspace relationship. The same types of performance breakdowns as investigated above are expected with other basic narrowband subspace methods.

In the next chapter, enhancements to the MUSIC algorithm are proposed to reduce the time required for computing a high-resolution result and to resolve ambiguous DOAs in excessively sparse arrays.

CHAPTER 4:
MUSIC Performance Enhancements and Limitations

4.1 Computational Performance

One of the major limitations of the MUSIC algorithm is that the search over the angular visible range is computationally intensive. The angular step size utilized in that search contributes to DOA error, so a small angular step size is desirable. A small step size means that there will be many angular values for which to test the orthogonality via (2.21).

4.1.1 Computational Complexity

Given a collector with a p-element receiver array, the array covariance matrix will be $p \times p$, and the eigenvectors will be of length p. The number of noise eigenvectors ($\mathbf{E_w}$) will depend on the number of signals; in general there will be $p-q$ noise eigenvectors. The steering vectors ($\mathbf{a}(\theta)$ will also be of length p. Neglecting any computations to derive the steering vector components and assuming that the steering vectors are already normalized, we only need to account for the matrix multiplications in the denominator of (2.21). Defining complexity as the number of floating point operations and assuming that both addition and multiplication are complexity order one $O(1)$, the complexity of a multiplication of two matrices of sizes $m \times n$ and $n \times k$ is $O(mk(2n-1)) \approx O(2mnk)$. The approximate complexity of the MUSIC algorithm for a single DOA is then:

$$\begin{aligned} O(MUSIC) =& O\left[\mathbf{a}^H(\theta)\mathbf{E_w}\mathbf{E_w}^H\mathbf{a}(\theta)\right] \\ =& O\left[(1\times p)(p\times(p-q))((p-q)\times p)(p\times 1)\right] \\ =& O\left[2p(p-q)+2p(p-q)+2(p-q)\right] \\ =& O(4p^2-4pq+2(p-q)) \\ \approx& O(4p^2-4pq) \end{aligned} \tag{4.1}$$

The number of receive elements will generally be relatively small, especially for a sparse array designed for aerospace applications; therefore, the leading scalars and the second order terms are retained because they will still have a large impact on the result. As an

example, for the six-element array of (3.1) with a single signal present the complexity is $\sim O(120)$.

This portion of the computational complexity is determined by the MUSIC algorithm; reducing it would require use of a different direction finding algorithm. The number of discrete angles to search and the angular step size between them is at the discretion of the system designer, however, and this provides an opportunity for improving computational performance. In general, if an angular resolution of Δ is desired, the number of discrete angles over which to conduct a one-dimensional MUSIC search in azimuth is $m = \alpha/\Delta$, where α is the angular width of the visible range. In the analyses of Chapter 3, α is 180°, or two in sinespace. For a direct implementation of MUSIC then, the complexity order for determining the full DOA spectrum would be $O\left(m\left(4p^2 - 4pq\right)\right)$. For practical applications, the number of discrete angles m will generally be much larger than the number of receivers and will be the primary driver of the number of computations required, especially when a two dimensional (azimuth and elevation) solution is desired. Therefore, reducing the number of discrete angles can be of great advantage to improving processing speed.

4.1.2 Reducing Search Complexity

One method of reducing the number of discrete angles without also reducing the resolution would be to run the MUSIC algorithm more than once. On the first run, a large step size could be used which provides only a rough estimate of signal direction. Those estimates can then be used to re-run MUSIC with a much smaller step size over only the angles surrounding each of the signal direction estimates, with the result of obtaining a high resolution DOA estimate without the large m which would normally be required. The question arises then of how best to choose a course resolution step size.

Since the ideal MUSIC DOA peak is infinite, a logical decision is to choose a step size which will provide a "good enough" peak height to distinguish signal DOA peaks from anomalous peaks which arise due to array sparsity in a linear array. Examination of the various MUSIC spectra plotted in Chapter 3, and especially that provided in Fig. 11(c) indicate that, at least for the array layout of (3.1), the anomalous peaks generally do not exceed a 10 dB orthogonality. While this is far from an exhaustive evaluation and is only based on trials with three or fewer simultaneous signals, it is acceptable for the purposes of

this discussion. An appropriately separated signal peak may then be defined as having an orthogonality of at least 20 dB. A step size can be derived from this by measuring the width of a signal's high resolution MUSIC response at a 20 dB orthogonality. The width and height of the MUSIC response of wideband signals can vary depending on the bandwidth and DOA but will never be narrower than the response for a sinusoidal signal because the minimum width is related to the array beamwidth, as indicated in the results Section 3.1.1. Therefore, the width of a sinusoidal signal at 20 dB orthogonality provides an acceptable metric for a coarse step size, Δ_{crs}. A fine resolution DOA can then be obtained by running (2.21) a second time with a fine step size over the range of $[-\Delta_{crs}/2, \Delta_{crs}/2)$.

A portion of the MUSIC DOA spectrum in a fine and two coarse angular step size for two sinusoidal signals at -29° and -12° is provided in Fig. 25. The fine step size is 2×10^{-4} in sinespace ($\sim 0.01^\circ$ at boresight), and the first coarse step size is 0.01 in sinespace (0.6° at boresight), which is the width of the fine step MUSIC response at 20 dB. The ratio of those step sizes is 50:1; while the exact advantage of using a two-step approach will depend on α and the number of signals present, a nearly 50 times speed improvement is highly desirable. As an example, when $\alpha = 2$ in sinespace and two signals are present, the fine step MUSIC requires 10,000 executions of (2.21), and the two step approach would require only 300, which is 33 times faster with the same signal DOA resolution! A third step size of 0.025 is also provided in Fig. 25 which indicates that as the step size increases, a point is reached where the MUSIC peak locations may no longer be resolvable.

It is also worth looking at the geolocation error induced by these step sizes. For an aircraft at 30,000 ft the horizon is $\sim$180 nmi away. An angular step of 2×10^{-4} implies a location uncertainty of $\pm$110 ft at that distance, while an angular step of 0.01 implies a location uncertainty of $\pm$0.9 nmi. If many collections of the same signals from the same direction were able to be completed with varying step sizes or step size offsets, this error could be averaged out. The more realistic scenario is that only a few signal collections may be likely from different positions as the collector moves, a situation in which use of a coarse step size could result in a significant location bias error.

While not a reduction in search complexity, it should be noted that the MUSIC algorithm is ideal for parallelization, which can reduce computation time when multiple computational cores are available or when the process is being embedded in a field programmable

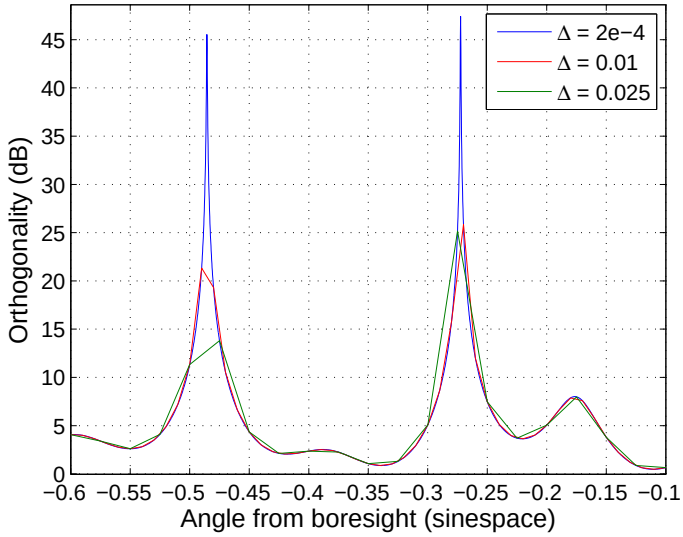


Figure 25. MUSIC DOA spectrum with varying angular step size.

gate array (FPGA). Since the MUSIC algorithm is a search over DOAs and the results for each discrete DOA are independent, each available core can be tasked with calculating a portion of the visible range. There is also no particular limit to the number of cores over which the process could be distributed aside from the number of discrete angles.

4.2 Ambiguity Resolution

In his paper on the MUSIC algorithm, Schmidt notes that there is an inability to resolve type I ambiguities, where $a(\theta_1) = a(\theta_2)$ and $\theta_1 \neq \theta_2$ [2]. This is true for a stationary collector and emitter, but ambiguity resolution can be accomplished when the collector is moving at a known velocity and multiple collections of the same signal can be obtained. This is because the true DOA will always point in the direction of the emitter, but any ambiguities will tend to drift.

To illustrate this ambiguity resolution ability, the array layout of (3.1) has been doubled so that the minimum spacing is now one wavelength:

$$d = \begin{bmatrix} 0 & 1 & 3 & 7 & 12 & 20 \end{bmatrix} \text{ wavelengths.} \tag{4.2}$$

This array spacing now has a grating lobe in addition to the main beam, which is illustrated

in the array radiation pattern plot of Fig. 26, where the main beam is at 10° and and the grating lobe is at 55.7°. This grating lobe manifests as a DOA ambiguity; there is no method of determining from a single collection which is the correct DOA. It should be noted that in sinespace the mainlobe/grating lobe separation will always be the same regardless of the mainlobe's offset angle from broadside; this is not true when plotting the radiation pattern in degrees. For a minimum receiver spacing of one wavelength, the lobes will be separated by one in sinespace.

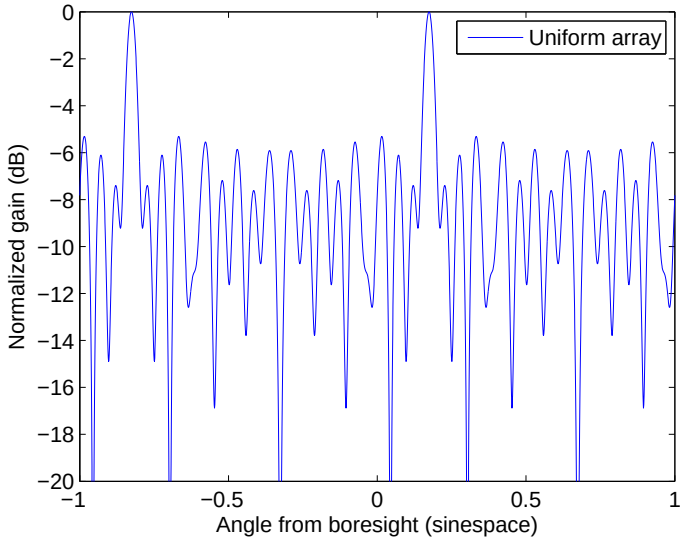
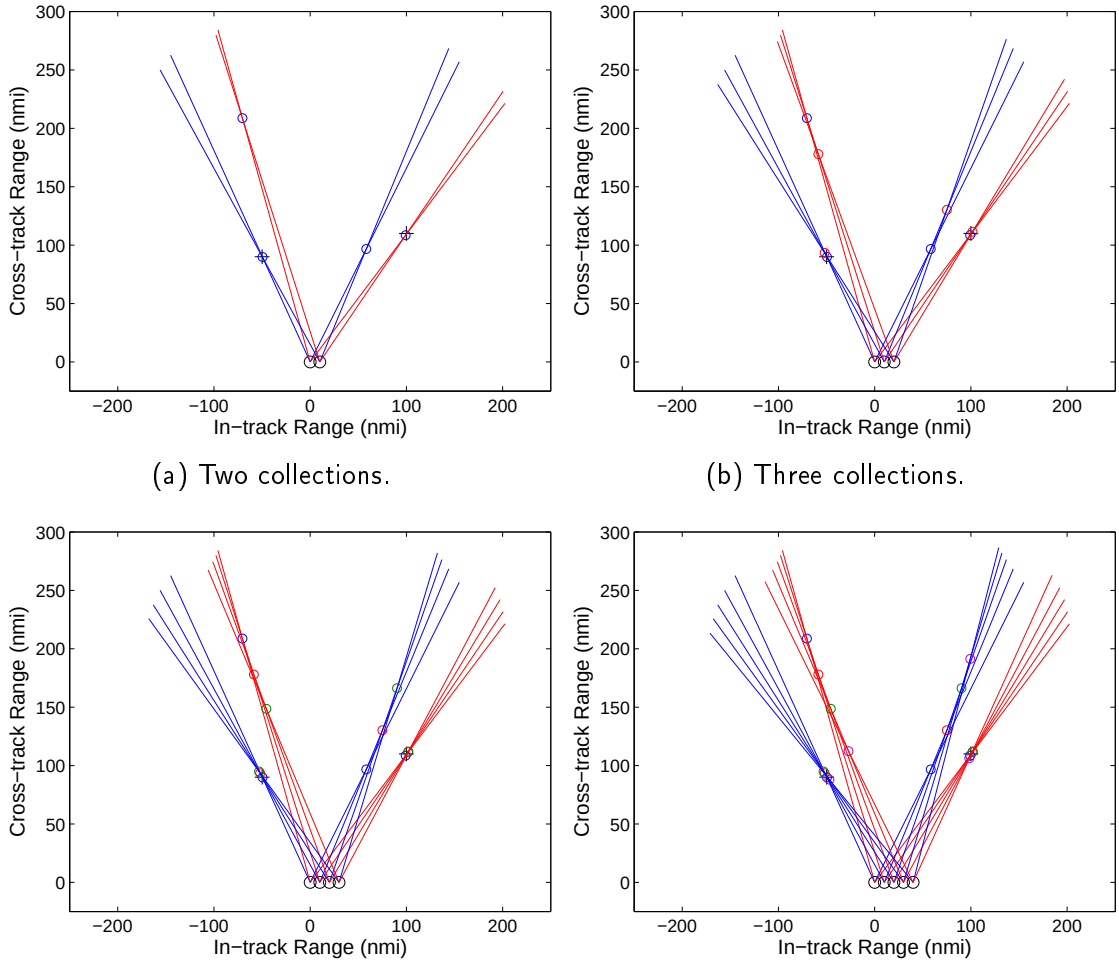


Figure 26. Radiation pattern for array with main beam steered to 10° and grating lobe at 55.7°.

To make an estimate of emitter location, all DOA vectors collected for a particular signal are utilized simultaneously. To determine DOA drift, however, only two need to be examined—the current and previous. By keeping track of the intersection points of DOA vectors from subsequent collections, it should easily be determinable that one set of crossing points remains relatively fixed while another set drifts.

An illustration of this principle is provided in Fig. 27, with a depiction of 2–5 collections of two 0 dB SNR sinusoidal signals near f_c. Each signal is a different color, and the two DOA vectors for each signal at each collection represent the two possible DOAs, only one of which is actually correct. The circles represent the DOA intersections of each subsequent collection.



(a) Two collections.

(b) Three collections.

(c) Four collections.

(d) Five collections.

Figure 27. Ambiguity resolution by tracking of DOA intersection points.

From examination of Fig. 27, it can be seen that the intersection point of one set of DOA vectors remain fairly constant, while the intersection points of the other set drift fairly significantly. The true DOAs are easily distinguishable by five collections, and a reasonable guess can be made with only three collections.

The MUSIC DOA spectra for the previous example are provided in Fig. 28, and it can be seen that the spectrum over sinespace $[-1,0)$ is an exact replica of that over sinespace $[0,1)$. This is due to the one wavelength minimum receiver separation in the array and the sinespace one separation in antenna lobes. When an ambiguity is present in the array,

one no longer has to process the entire visible range since each signal will be repeated more than once in the MUSIC spectrum. This does not necessarily equate to halving the processing time; since the antenna beamwidth is now narrower, the MUSIC response is narrower, and a smaller step size may be desired to ensure an appropriate peak amplitude can still be determined.

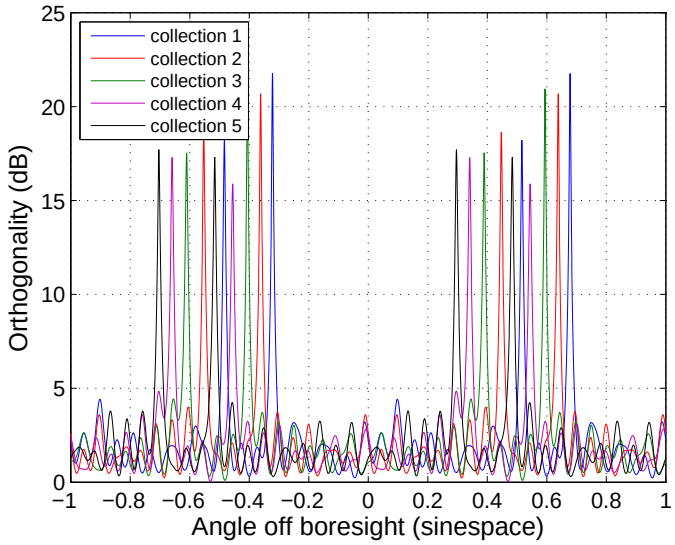


Figure 28. MUSIC DOA spectra for receiver array with single ambiguity.

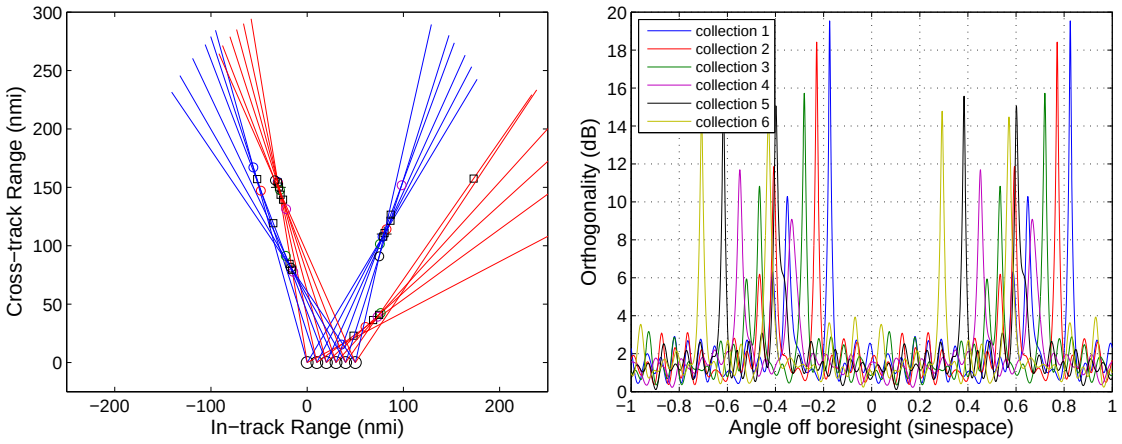
A metric for comparing the motion of intersections between DOA possibilities can be calculated by determining the vector between any two successive intersections, summing all of these vectors for each DOA, and determining the square of their length. This is also equivalent to just determining the magnitude of the vector from the first to the last collected intersection points. The metric for the true DOA should be much smaller than that for any ambiguous DOAs since the ambiguous DOA intersection points will travel further. The metric values for the prior example are provided in Table 7. By these metrics, for Signal 1 (blue in Fig. 27) the negative DOA is correct, and for Signal 2 (red in Fig. 27) the positive DOA is correct, which matches the graphical interpretation.

As shown in Chapter 3, high bandwidths and signals of different power can result in various sorts of DOA estimation degradations, of which the severity is dependent also on the exact DOAs and frequencies present. These degradations lead to DOA estimation errors, which can lead to the intersection points of subsequent signals not to drift in an orderly fash-

Table 7. Ambiguity metric over multiple collections with sinusoidal signals.

Signal	DOA	Collection 3	4	5
Signal 1	DOA 1	185	391	10
	DOA 2	1408	5869	10646
Signal 2	DOA 1	1093	4244	11191
	DOA 2	152	205	6

ion or to jitter around the true emitter position. A slightly more robust ambiguity metric can be calculated using the midpoint of two subsequent intersection points vice the intersection points themselves. One downside is that this requires at least four collections before making a judgment on which is the correct DOA. An illustration of the map view and MUSIC spectra for six collections with a 15 percent bandwidth LFM and QPSK signal, both at 10 dB SNR, are provided in Fig. 29(a). Circles again indicate the intersection points and squares indicate the mid-position of subsequent intersection points. The ambiguity metrics calculated using both the DOA intersection points and the intersection point mid-positions are provided in Table 8, and a ratio of the metric for the true DOA divided by the metric for the ambiguous DOA is provided for easy comparison between the two methods. In general, the mid-position metric provides a better estimation and has less variability.



(a) Map view. (b) MUSIC DOA spectra.

Figure 29. Ambiguity resolution for receiver array with single ambiguity and wideband signals.

Table 8. Ambiguity metrics over multiple collections with wideband signals.

Signal	Point type	DOA	Collection 3	4	5	6
Signal 1	Intersection	DOA 1	456	6857	9708	8878
		DOA 2	58	47	2334	293
		Ratio	0.127	0.007	0.240	0.033
	Mid-position	DOA 1	-	1714	6402	7349
		DOA 2	-	12	289	146
		Ratio	-	0.007	0.045	0.020
Signal 2	Intersection	DOA 1	1	54	624	10
		DOA 2	852	2397	1975	123985
		Ratio	0.001	0.023	0.316	0.000
	Mid-position	DOA 1	-	14	245	119
		DOA 2	-	599	1032	33683
		Ratio	-	0.023	0.237	0.004

One issue that arises with arrays with ambiguities is that the true DOA of one signal could be co-incident with an ambiguous DOA of another signal. In this case, the MUSIC response will only have one peak in that location, which could make separating the two signals difficult or impossible.

4.3 Limitations

While some enhancements and understanding of the MUSIC algorithm are possible, there are some fairly significant limitations which make real-world implementation difficult beyond the computational complexity discussed in Section 4.1.1.

4.3.1 Steering Vector Estimation

Equation (2.8) is an analytical estimate for a steering vector based on the frequency, angle, and receiver mounting locations, but for a real array the steering vectors will not be linear functions of frequency and angle because of mutual coupling of receive antenna elements and effects of the vehicle structure to which the elements are mounted. For a real implementation, the steering vectors will generally be determined via array calibration and stored as a lookup table [2]. It would be impractical to measure and store a lookup table with very small steps in both frequency and angle: for an array with a 180° visible range, a 1 GHz bandwidth, a frequency step of 100 Hz and an angular step of 0.1°, a lookup table

with 1.8×10^8 entries would be required—and that is only for azimuth determination!

Errors in receive element positioning and pointing are also unavoidable, and even disregarding mutual coupling, the radiation patterns on multiple elements will not be identical. All of these factors combine to create errors in one's knowledge of the steering vectors which will result in additional DOA estimation errors.

4.3.2 Signal Number Estimation

One limitation that has been ignored in this paper is in estimating the number of signals q present during any collection period. Knowing this is essential to determining which of the array covariance matrix eigenvectors are signal eigenvectors and which are noise eigenvectors, because only the noise eigenvectors are used in the MUSIC algorithm of (2.21). This limitation is not specific to the MUSIC algorithm but exists for all subspace algorithms.

In Section 2.4, it is noted that the $p - q$ smallest eigenvalues should all equal σ_w^2, but in practice eigenvalues will fall along a continuum and it can be difficult to distinguish the smallest signal eigenvalue from the largest noise eigenvalue. There are numerous items in the literature which develop and analyze statistical likelihood ratio tests which attempt to solve this problem. Wax and Kailath wrote one of the seminal papers on the subject with a development and analysis of estimators based on two information theoretic criteria termed Akaike information criterion (AIC) and minimum descriptive length (MDL) [20]. Hill and Pickholtz later provided a Monte Carlo-based calculation approach to compare the effectiveness of a number of different estimator types [21], and more recently Nadakuditi and Edelman developed a new method simply referred to as the "new" method [22].

These methods are all functions of the number of receivers, the number of samples, and the eigenvalues of the array covariance matrix. Code to implement two of these methods, Wax's MDL and Nadakuditi and Edelman's "new" method, is contained in Appendix B. For the analyses in this paper, however, both estimators nearly always returned the maximum possible number of signals (five in the case of a six-element array) and were almost never correct. Due to time constraints further research on the reasons for this were not attempted. For this reason the estimated number of signals was set to equal the known number of signals for all implementation of the MUSIC algorithm employed in the examples and

analyses of this paper.

4.3.3 Signal Matching

Localization of a particular signal over multiple collections requires that the signals collected at different times can be identified as being the same signal. With MUSIC and other subspace algorithms, if more than one signal is present at one time the incident signals can only be estimated by reconstruction (see Section 2.4.2) using the estimated DOAs. Even with sinusoidal signals, reconstruction is only conducted using a limited number of samples, and two collections will not match exactly. With radar pulses it may be particularly difficult to tell the difference between two emitters operating at the same frequency with such a limited sample set. Additionally, the blending that can occur with wideband signals (see Section 3.3) creates a severe impediment to accurate signal matching since the effect will be different for each combination of signals and DOAs.

4.4 Performance Enhancements and Limitations Review

In this chapter, a method of reducing the number of computations required to obtain a high-resolution DOA estimate was proposed; this method utilizes a two-step execution of the MUSIC algorithm to first obtain a coarse DOA spectrum over the entire visible range and then to calculate a fine DOA spectrum over very limited portions in the vicinity of the signal DOAs. A method of resolving ambiguities for moving receiver arrays with minimum receiver spacing $> \lambda/2$ was also proposed, and two metrics were derived which can provide an estimate of the true DOA within three or four collections. Finally, some of the limitations inherent in the MUSIC algorithm and in this research were delineated.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Conclusion

5.1 Results

This thesis research was conducted with three main goals: to implement the MUSIC subspace direction finding algorithm and geolocation in MATLAB; to conduct a high-level performance analysis of the algorithm with particular emphasis placed on the effects of frequency, bandwidth, DOA, and SNR and to devise some enhancements to the process, if possible.

5.1.1 Performance Analysis

In the context of this research, performance is defined as the ability to resolve a signal's DOA and the accuracy of that DOA. It has been shown that signals not at the array center frequency will have a shifted response in the MUSIC DOA spectrum, but the frequency alone does not result in a degradation of the response. It has also been shown that the weighted mean frequency, described in Section 3.2, provides an acceptable method of estimating and correcting the MUSIC DOA shift. The accuracy of the subsequent DOA determination is then based on the accuracy of the reconstructed signal's frequency. For wideband signals, it was shown in Section 3.3 that signal blending can occur during reconstruction; this blending will corrupt the frequency estimation and the DOA estimation. Over multiple collections there may be situations where a signal-of-interest is blended with different signals on each collection; it is expected that this will result in a colored noise effect on the signal DOA estimates and the subsequent geolocation.

Besides the effect on signal reconstruction, it was demonstrated that the MUSIC spectrum peak amplitudes decrease roughly linearly with increased signal bandwidth. Because a peak still exists even at extremely high bandwidths of 40% f_c, this reduction in amplitude might at first seem to be immaterial; however, when combined with an offset in DOA from array boresight, the amplitude reductions can be significant. Additionally, if multiple signals are present there will be a reduced ability to resolve two closely spaced signals and there will be a higher likelihood of distortion due to the presence of multiple signals.

An example of both of these effects was presented at the end of Section 3.3. While there are an infinite number of possible signal combinations and an exhaustive analysis was not conducted, this example indicates that 5–10 percent signal bandwidth is the effective limit for the baseline MUSIC algorithm to maintain signal resolution and avoid DOA distortion. The evaluation of the effect of DOA on the MUSIC spectrum in Section 3.4 also indicates that even with a bandwidth of $\leq 10\%\ f_c$, MUSIC is likely to only be truly effective out to $\sim 30^\circ$ off boresight. This result is interesting also in that an array is likely to be comprised of receiver elements which have some directivity and do not have a perfectly semicircular pattern in azimuth. It may make sense to choose receive elements with beamwidths which effectively match the DOA degradation for the expected bandwidth of received signals. For the situation described above, receivers with an approximate gain of 9 dB and a 3 dB beamwidth of $\sim 60^\circ$ could be utilized. If one is willing to sacrifice the ability to detect narrowband signals beyond 30° off boresight, a computational benefit can also be realized in reducing the MUSIC search to only $\pm 30^\circ$. Since 30° is 0.5 in sinespace, this is effectively a halving of the number of computations required as compared to a search over $\pm 90^\circ$.

An analysis of the effects of signal SNR was presented in Section 3.5, and it was shown that MUSIC is still fairly accurate even down to -10 dB SNR, though the peak amplitude is reduced. Monte Carlo simulations indicate that the effect of noise is unbiased over multiple collections, but that the accuracy of a particular collection DOA is degraded to a standard deviation of approximately a quarter of a degree at -10 dB SNR. Additional effects are discernible when multiple signals are present. First, as SNR decreases, there is a reduction in the angular resolution between source signals. Second, an investigation of the MUSIC response with overlapping signals of different power indicates that, especially with wideband signals, there is an effective limit of 10 dB difference between signal powers in order to be able to resolve the presence of the weaker signal in the MUSIC DOA spectrum.

These results will prove useful to system designers considering implementation of the MUSIC algorithm. If the system will be used in an environment where signal overlap is expected, source signals have a 5% bandwidth or less, and signal powers do not vary greatly, then the basic MUSIC algorithm should perform quite well. Between 5–10% bandwidth and depending on the number of concurrently overlapping signals, the basic MUSIC algorithm may not provide the desired accuracy, and as bandwidths of 20% or greater are

approached an alternate method which can correct for the bandwidth effects is likely required.

5.1.2 Enhancements

The first performance enhancement developed in this thesis is a method of reducing computational complexity through the use of a two-stage DOA determination. In the first stage a coarse angular step size is utilized for the MUSIC search; the step size should be chosen to ensure a minimum peak amplitude for signal detection. The second step runs the exact same MUSIC calculation, but this time with a fine step size which should be chosen based on desired geolocation precision. The second stage is only run in the vicinity of the signal peaks which were extracted from the first stage. In this manner a high resolution can be obtained with significantly fewer calculations than if the fine step size were to be utilized over the entire visible range. The example developed in Section 4.1 resulted in an approximately 30 times speed improvement. This improvement enables the real-time implementation of the MUSIC algorithm in less powerful computing hardware than might otherwise be required or available.

A geolocation enhancement was also introduced in Section 4.2. Generally, arrays with ambiguities, or grating lobes, are undesirable for direction finding applications. The motion of an airborne or spaceborne array provides a method of resolving those ambiguities, however, under the assumption that the same signal can be received and correlated over multiple collection durations. This is possible because the true DOA will always point towards the transmitter, but the false DOA (s) will drift and not all cross in the vicinity of a single point. A DOA drift metric is also derived which enables automatic estimation of the correct DOA with a minimum number of collections. This enhancement enables the construction of arrays where receiver size or other constraints to the minimum receiver spacing result in the presence of unavoidable grating lobes.

5.2 Future Work

There are many areas in which this thesis work could be expanded. The most immediate research need would be in an examination of some of the more recent and more advanced subspace methods which could reduce computational complexity and reduce the impact of signal bandwidth on the determination of signal DOA. Friedlander and Weiss's work in

using virtual arrays [23] should enable a reduction in computational complexity by enabling the use of root-MUSIC [7], but the accuracy of estimating interpolated signal phases for a sparse array would need to be closely examined, as would the complexities arising from the use of multiple virtual arrays over different portions of the visible range. There have been multiple papers written since Schmidt's seminal paper on MUSIC [2] concerning methods to improve the subspace DOA resolution and minimize interaction effects for wideband signals; a primary candidate for future research would be the method developed by Friedlander and Weiss which builds on top of their prior work on virtual arrays [11]. A comparison to the methods of Doran, Doran, and Weiss [10] as well as other promising methods would also prove fruitful.

There are also multiple possible avenues of directly expanding on the work done in this thesis. One area would be in an evaluation of the impact of collection duration on the MUSIC spectrum; at a minimum an improvement in DOA resolution at low SNRs is expected, but there may be other advantages and disadvantages as well. The implementation of more realistic signals would confirm that methodology employed by this thesis of using representative test signals was appropriate; this should include more realistic signal durations and power levels. Examples could include phase-coded radar signals such as maximal-length and Frank polyphase codes or communications signals such as 16-QAM using Nyquist pulse shapes. Finally, the impact on DOA estimation and signal reconstruction should be evaluated with irregularly timed signals which break across collections.

Secondly, a deeper investigation into the methods of signal number estimation and their limitations could resolve the reasons why this thesis research was unable to obtain correct estimations. Being able to make an estimation with at least a reasonable expectation of accuracy is an essential piece of realizing an operational employable subspace-based direction finding system. This research should begin with the papers in [21] and [22] and include any other relevant work in the area. At a minimum, the impact of signal type, bandwidth, SNR, and array layout on the ability to estimate the number of signal should be investigated.

A third research area would be in methods of improving the signal reconstruction to avoid the blending which can occur with wideband signals, as demonstrated in Section 3.3. An analysis of the ability to correlate separate collections as the same source signal in the

presence of blending could also prove useful. This is of particular concern in high-density signal environments where it can be especially difficult to separate and geolocation signals.

A final possible field of study would be in applying the model error analysis of Swindlehurst and Kailath [13] for non-linear arrays. Additionally, their weighting scheme for MUSIC could be evaluated to see how it performs for the same radar-type signals utilized in this thesis over a range of frequencies and bandwidths.

5.3 Conclusion

It is hoped that the results of this thesis, which are a first step towards deeper investigations into the subspace direction finding algorithms, will enable an expansion of the geolocation research which has been on-going at the Naval Postgraduate School to include more analysis beyond the more traditional two-receive element geolocation methods. The ability to accurately utilize MUSIC in moderate to low bandwidth environments, despite it being derived specifically for sinusoidal signals, was also demonstrated. Additionally, the density of RF emitters is increasing around the world, and with that comes an increase in the likelihood of receiving overlapping signals. Methods of accurately resolving and geolocation those overlapping signals are likely to be of interest to the military in the future, which means that now is the time for research into subspace direction finding and other promising methods of multiple signal geolocation.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A:
Subspace Processing Specifics

A.1 Time-Domain Steering Matrix with Wideband Signals

The time-domain solution derived in Section 2.4, and specifically the definition of the steering vectors as (2.8), is only an exact relationship for sinusoidal signals. To show the true nature of the steering matrix, it is useful to derive an analytical solution for the representative wideband signals utilized in this thesis.

Let $s_1(t)$ be a LFM pulse at center frequency f_{c1}, bandwidth B, and duration T, and let $s_2(t)$ be a phase-coded signal of frequency f_{c2} and phase sequence $\psi(t)$:

$$s_1(t) = e^{j2\pi\left[\left(f_{c1}-\frac{B}{2}\right)t+\frac{B}{2T}t^2\right]} \tag{A.1}$$

$$s_2(t) = e^{j(2\pi f_{c2}t+\psi(t))}. \tag{A.2}$$

Additionally, assume that the two signals arrive at time zero at the origin of a three-element array. There will be no phase shift at the origin, but the phase shift of signal s_i at a receive element offset d_j from the origin will be in accordance with (2.1). For this derivation, it makes more sense to express the phase in terms of frequency, where c is the speed of light:

$$\phi_{ij} = 2\pi f_i \frac{d_j}{c}\sin\theta_i. \tag{A.3}$$

The DOA θ_i and the receiver offset from the origin d_j are both straightforward, but the frequency f_i is not. The instantaneous frequency of a signal is the derivative of its phase in cycles; for the two signals in (A.1) and (A.2) the phase in cycles is

$$\phi_1(t) = \left(f_{c1}-\frac{B}{2}\right)t + \frac{B}{2T}t^2 \tag{A.4}$$

$$\phi_2(t) = f_{c2}t + \frac{1}{2\pi}\psi(t). \tag{A.5}$$

The instantaneous frequency of each signal is then

$$f_1(t) = \left(f_1 - \frac{B}{2}\right) + \frac{B}{T}t \tag{A.6}$$

$$f_2(t) = f_2 + \frac{1}{2\pi}\frac{\mathrm{d}\psi(t)}{\mathrm{d}t}. \tag{A.7}$$

It is clear that the instantaneous frequency of each signal is time-dependent, potentially non-linear, and will change depending on the specific frequency/bandwidth/phase parameters of each signal. Substituting (A.6) and (A.7) into (A.3) will result in a time-dependent steering vector. The constant steering vector of (2.8) is then simply an approximation which may only truly be accurate over a very short portion of the collection. It is from this inaccuracy that most of the MUSIC degradations explored in Chapter 3 arise.

A.2 Subspace Methods in the Frequency Domain

The array matrix relationship of (2.6) can also be developed in the frequency domain; doing this offers some computational advantages over the time domain method. At the origin, the received signal is the sum of all q incident signals, just as in the time domain, with no phase shifts:

$$X_0(f) = \sum_{i=1}^{q} S_i(f), \tag{A.8}$$

where $S_i(f)$ is the Fourier transform of the ith signal. For any particular source signal $S_i(f)$, there will be a phase shift in the received signal at a receiver offset from the origin, and that phase shift will be the same as it is in the time domain. In Section A.1, it was shown that the frequency component of the phase shift was time dependent for wideband signals and had to be derived separately for each signal. In the frequency domain, however, assuming that the DFT was utilized to obtain the frequency spectrum, the frequency term of the phase shift will simply be the vector of frequency bins from the DFT. This simpler and always linear expression of the steering vectors for wideband signals is the first advantage of conducting the subspace algorithms in the frequency domain. A second advantage is that even though there are the same number of frequency bins as time domain samples, it is relatively easy in the frequency domain to reduce the number of samples for which the algorithm needs to be run. The most obvious way to reduce the number of samples in the

calculation of the covariance matrix in the frequency domain is by using only the positive frequency bins. Since the subspace algorithms are run on the analytic signal, as long as the algorithm is executed at RF or IF the negative frequency portion of the real signal frequency spectrum can be discarded since it is simply a duplicate of the positive frequency portion. The number of covariance samples can be further reduced if one is only concerned about signals over a limited frequency band. The main potential issue with reducing the number of received signal samples in the frequency domain is that there are then fewer samples over which to average out the noise, and the noise in any particular frequency bin can have a larger impact on degrading the signal and noise subspace relationship.

In fact, since the Fourier transform is complex and can be expressed in magnitude and phase components, only the phase portion of the frequency spectrum is necessary to compute the array covariance matrix and determine signal DOAs. Since the received signal at an element offset from the origin is simply an additional phase shift, the magnitude of the received frequency spectrum is exactly the same at all receivers, neglecting noise and very small differences in distance from the source and assuming that all receivers have an identical radiation pattern.

Working in the frequency domain also means that if any signal frequency corrections are required (see Section 3.2), all of the required information to estimate signal center frequencies is already present and one does not have to switch back and forth between the time and frequency domains.

A.3 Subspace Method Processing at an Intermediate Frequency

Subspace method direction finding processing can be completed at an IF with the same results as if the processing were completed at RF, disregarding any non-linearities resulting from the mixing process and assuming that the received signal at all receiver elements is mixed down to the exact same IF prior to sampling. For the purposes of this discussion, f_r is the array center frequency at RF, f_m is the mixer frequency, and $f_i = f_r - f_m$ is the center frequency of the IF received signal.

When using the MUSIC algorithm at IF, while the array covariance matrix (2.12) is

computed using the IF signals, the steering vectors used in the DOA search of (2.21) must be computed using the RF array center frequency f_r. This same steering vector construction with f_r is used during signal reconstruction and frequency estimation. The resulting signal estimated frequency in this case will be at IF, but any frequency correction (see Section 3.2) must be completed as it would be for the RF signal by including an adjustment for the mixer frequency. Therefore (3.5) must be re-written as

$$\theta_{i,est} = \arcsin\left(\frac{f_r}{f_{i,est} + f_m} \sin\theta_{i,MUSIC}\right). \tag{A.9}$$

APPENDIX B: MATLAB Code

A working example of the MUSIC direction finding algorithm, along with test signal generation, estimated signal reconstruction, geolocation, and plotting of results is provided below. The enhancements discussed in Chapter 4 are also included in the primary script. A supplemental electronic copy of this code can be obtained by contacting the Naval Postgraduate School's Dudley Knox Library. The electronic version of the code also includes a script which generates the example and visualization contained in Chapter 2 and a script and associated signal definition files which generate the results of Chapter 3.

B.1 MUSIC Direction Finding and Geolocation Script

```
%% MUSIC direction finding algorithm and geolocation
%
% Purpose:
%   This script generates test signals which are defined by a specified
% file and runs the MUSIC algorithm in the frequency domain in azimuth
% only based on parameters sepcified in the initilization section below.
% Signal reconstruction based on the derived DOAs is also conducted, and
% if more than three collections are commanded an estimate is made of
% the emitter location.  Data is presented in plots for easier analysis.
%
% Custom functions called:
% sig_config_xxxx - not a function, but a script file which contains
%   specifications for one or more signals as well as the array center
%   frequency, sample frequency, and signal duration
% interferometry_sig_gen - function which generates phase shifted
%   signals at specified receiver locations
% subspace_calc - function which generates the array covariance matrix
%   and the eigendecomposition of said matrix.  Also estimates number of
%   signals if desired
% music_calc - computes MUSIC spectrum
% geolocate2 - 2D least squares estimate of emitter locations  based on
%   collected DOA vectors
```

```
% pt_motion - metric for ambiguity resolution; measures the motion of a
%   test point (currently intersection mid-positions) over multiple
%   collections
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
% NOTE
%   This is a full implementation of the code used for this thesis, but
% it does not directly generate the figures contained in Chap 3 of the
% thesis. Those were obtained through rearrangements and subplotting in
% order to highlight specific results; the script used for that is
% attached electronically to the NPS library's electronic version of
% this thesis.  The code used to generate the example and visualization
% in Chap 2 of the thesis is also attached electronically.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Initalization - User options

clear all
close all
%clc

arraytype = 'sparse';   % options are 'uniform' or 'sparse'
num_ambig = 0;          % 0 = no ambiguities, integer = # of ambiguities
noiseon = 0;            % Turn noise on or off.  1 = on, 0 = off
save_figs = 0;          % 1 = save figures, 0 = do not save figures

% compute signal number estimation.  Options are 'none', 'NEW', or 'MDL'
sig_num_est = 'none';

sig_config_example;      % load signal configurations from file
selected_sigs = [1 2];   % choose signals to model

% range of frequencies for array (Hz)
fmin = recParam.centerfreq - 500e6;
fmax = recParam.centerfreq + 500e6;
```

```
62
63 num_collects = 5;         % number of collection runs
64
65 %% Additional Setup
66
67 global c
68 c = 3e8;     % speed of light (m/s)
69
70 % Angles at which MUSIC spectrum will be computed (this is main driver
71 % of processing time).  Two-stage method is being utilized.
72 sine_step_crs = 0.001;        % Coarse step size (sinespace)
73 sinespace = -1:sine_step_crs:1-sine_step_crs; % discrete angles
74 angles = asind(sinespace);  % discrete angles (deg)
75 sine_step_fine = 2e-4;       % fine step size (sinespace)
76
77 % initializations
78 shift_req = 0;
79
80 % custom color order definition
81 colors = [   0         0    1.0000;
82         1.0000         0         0;
83              0    0.5000         0;
84         0.7500         0    0.7500;
85              0         0         0;
86         0.7500    0.7500         0;
87         0.2500    0.2500    0.2500];
88
89 %% Array Configuration
90
91 % Receive element locations; array spacing in the specified vector is in
92 % center frequency wavelengths
93 switch arraytype
94     case 'uniform'  % half wavelength spacing
95         rec_el_NL_wav = ([0 1 2 3 4 5]'/2);
96     case 'sparse'
97         % sparse spacing (minimum half wavelength spacing, no
98         % ambiguities, consistent sidelobe height of ~ -6 dB). Pattern
99         % is to place additional emitters at distances which equate to
100        % an unbroken string of integer half wavelengths when all
```

```
        % elements are considered relative to each other.  For the
        % current 6-element array, the half wavelength separations
        % covered are: 1-9, 11-13, 17, 19, 20. The first skipped integer
        % is 10, so the next element should be added 10 half wavelengths
        % after the last one.  Adding a 7th element at d=30/2 would add
        % the separations: 10, 18, 23, 29, 30.  Array can continue to be
        % expanded in this manner.
        rec_el_NL_wav = ([0 1 3 7 12 20]'/2);
    otherwise
        error('Incorrect array type specified')
end

% adjust array spacing for desired number of ambiguities
num_lobes = num_ambig + 1;
rec_el_NL_wav = rec_el_NL_wav * num_lobes;

% Receive element locations in meters
rec_el_NL = rec_el_NL_wav*c/recParam.centerfreq;
p = length(rec_el_NL);      % number of receive elements

% Steering vector definition.  Defined as a function for later
% flexibility
Af = @(f,ss) exp(-1j*2*pi*f*rec_el_NL*ss/c);

% Determine number of ambiguitities at center frequency
phi = 2*pi*rec_el_NL_wav*sinespace; % matrix of receiver phases by angle
v = exp(1j*phi)/sqrt(p); % complex phases, corrected for array size

% normalized complex antenna pattern at specified angle. shift to
% 360*3/7 deg (increases peak finding accuracy); using a large enough
% prime number denominator (7) minimizes the chances of a peak wrapping
% around at +-90 deg (which is difficult to count properly); the
% numerator (3) servers to more or less recenter the main beam.
pattern = v(:,ceil(length(sinespace)*3/7))'*v;
pattern_mag = abs(pattern);
% ambiguity is being defined as > 90% of main beam magnitude.  The MUSIC
% response for grating lobes below 90% is small.  This step truncates
% the data at 0.9
pattern_mag(pattern_mag<=.9)=0;
```

```
[pks, locs] = findpeaks(pattern_mag);
num_lobes = length(pks);   % count number of peaks remaining

%% Collector details

coll_pos = [0; 0];     % initial collector position

coll_vel = [300; 0];       % nmi/hr
coll_vel = coll_vel/3600;     % nmi/s

collect_interval = 120;  % collection interval, in seconds

% rotation matrix to normalize relative DOAs
% (not currently utilized)
R = 1/norm(coll_vel)*[coll_vel(1),-coll_vel(2);coll_vel(2),coll_vel(1)];

%% Collection Loop

for n = 1:num_collects
    % collection time (s); 1st collection defined as t = 0
    t_col(n) = (n-1)*collect_interval;
    % collector position
    coll_pos(:,n) = coll_pos(:,1) + coll_vel*t_col(n);

    % signal generation
    [sigs, X, e_pos, sig_names, doas(:,n), N, t] =...
    interferometry_sig_gen(sigParam, recParam, selected_sigs,...
    coll_pos(:,n), coll_vel, rec_el_NL, noiseon);

    q = length(selected_sigs);  % number of source signals

    Xf = fft(X,[],2);       % convert to freq domain
    df = ((0:(N-1)).*(recParam.sampfreq/N))';   % FFT frequency vector

    % Subspace determination
    [q_est, EigVec_s, EigVec_n, EigVal] = subspace_calc(Xf,...
        sig_num_est, q, n);

    %% MUSIC
```

```
179
180     % Run MUSIC with coarse steering vectors
181     music_spectrum(n,:) =...
182         music_calc(Af(recParam.centerfreq,sinespace), EigVec_n);
183
184     % define the number of desired peaks to equal the number of
185     % estimated signals times the number of ambiguities
186     % NOTE - This assume that each true DOA has a grating lobe, which is
187     % only exactly true for integer multiples of minimum wavelength
188     % spacing
189     num_peaks = q_est*num_lobes;
190     % extract peaks from MUSIC plot in descending order
191     % NOTE - high bandwidth signals can lead to an anomolous peak being
192     % chosen over a true peak.
193     [pks, locs] = findpeaks(real(music_spectrum(n,:)),'sort','descend');
194
195     % re-sort q_est tallest peaks into DOA order (negative to positive),
196     % store as coarse angle vector
197     ambig_locs_crs_ss{n} = sinespace(sort(locs(1:num_peaks)));%sinespace
198     ambig_locs_crs_ang{n} = angles(sort(locs(1:num_peaks)));  % degrees
199
200     % re-run music calculation with higher precision steering vectors
201     for m = 1:num_peaks;
202         % angle around which to conduct higher resolution search
203         peak_ang = ambig_locs_crs_ss{n}(m);
204         % define fine resolution segment of visible range
205         sine_sub_ang = (peak_ang-sine_step_crs/2):...
206             sine_step_fine:(peak_ang+sine_step_crs/2);
207         % re-run MUSIC
208         music_hires(n,:) =...
209             music_calc(Af(recParam.centerfreq,sine_sub_ang),EigVec_n);
210         % determine appropriate higher resolution DOAs and store as
211         % precise angle vector
212         % NOTE - currently written to only look for one peak
213         [~, indx] = max(real(music_hires(n,:)));
214         ambig_locs_p_ss{n}(m) = peak_ang - sine_step_crs/2 +...
215             indx*sine_step_fine;
216     end
217     % convert to degrees
```

```
218     ambig_locs_p_ang{n} = asind(ambig_locs_p_ss{n});
219
220     % fit DOA estimate into an ambiguities x signals matrix
221     doa_est_ss{n} = reshape(ambig_locs_p_ss{n},q_est,num_lobes).';
222
223     %% frequency Domain Signal Reconstruction
224
225     % use array center frequency and derived DOAs to solve for an
226     % estimate of the source signals
227     SIG_est{n} = pinv(Af(recParam.centerfreq,doa_est_ss{n}(1,:)))*Xf;
228     % convert to time domain
229     sig_est{n} = ifft(SIG_est{n},[],2);
230
231     %% Frequency Estimation
232
233     % create normalized PSD for each estimated source signal
234     SIGPWR_est_norm = (diag(1./...
235         max(abs(SIG_est{n}(:,1:ceil(N/2))),[],2))*...
236         abs(SIG_est{n}(:,1:ceil(N/2)))).^2;
237
238     % truncate lowest 10% of response (minimizes noise and effects of
239     % signals not being centered in search bandwidth)
240     SIGPWR_est_norm(SIGPWR_est_norm<.1) = 0;
241
242     % estimate signal center frequency
243     f_sig_est{n} = ((SIGPWR_est_norm*df(1:ceil(N/2)))./...
244         sum(SIGPWR_est_norm,2)).';
245
246     %% MUSIC adjustments
247
248     % fix DOAs based on incorrect frequency assumption, uses fft
249     % estimated center freq from above
250     doa_est_ss_ff(:,:,n) = recParam.centerfreq./...
251         repmat(f_sig_est{n},num_lobes,1).*doa_est_ss{n};
252
253     % Determine direction vector to emitter
254     % TBD: and correct for movement not in the +x direction
255     % w = {# coll}(sig #, ambig #, (sin(DOA)|cos(DOA))
256     % Note that (sin(DOA),cos(DOA)) is an (x,y) unit vector in the
```

```
257     % direction of the emitter (assuming +x motion)
258     w{n}(:,:,1) = doa_est_ss_ff(:,:,n); % x components
259     w{n}(:,:,2) = sqrt(1-w{n}(:,:,1).^2); % y components
260
261     %% geolocation determination
262     % NOTE - difficult to functionalize as currently written)
263
264     if n > 1
265         % intersection point of two DOAs
266         intersect_pos{n} = intersect_pt(coll_pos,w);
267         if n > 2
268             % if no ambiguities, solve geolocation problem
269             if num_lobes == 1;
270                 doa_est_ss_ff_na = reshape(doa_est_ss_ff,q_est,n).';
271                 e_pos_est = geolocate2(coll_pos, doa_est_ss_ff_na);
272             else
273             % if ambiguities, attempt to determine correct DOA
274                 mid_pos{n} = (intersect_pos{n} + intersect_pos{n-1})./2;
275                 if n > 3
276                 % note: length(sum(distance vectors) has more dynamic
277                 % range than sum(length(distance vectors).  Also, use of
278                 % mid-position has better results than individual
279                 % crossing points (though the latter can be calculated
280                 % one iteration sooner)
281                     sq_mid_motion{n} = pt_motion(mid_pos);
282                     % min values are likely correct DOA (not ambiguity)
283                     [~, indx] = min(sq_mid_motion{n},[],1);
284                     for m = 1:q_est
285                         if n == 4
286                             % go back and remove ambiguous DOAs from
287                             % first 3 collections
288                             for k = 1:3
289                                 doa_est_ss_ff_af(k,m) =...
290                                     w{k}(indx(m),m,1);
291                             end
292                         end
293                         % remove ambiguous DOA from current collection
294                         doa_est_ss_ff_af(n,m) = w{n}(indx(m),m,1);
295                     end
```

```
296                     % Estimate emitter position using estimated correct
297                     % DOA
298                     e_pos_est = geolocate2(coll_pos, doa_est_ss_ff_af);
299                 end
300             end
301         end
302     end
303 end
304 
305 %% plots
306 
307 % determine if signals 'wrap' around +-90 deg; shift DOAs if necessary
308 % (note - cumulative if there's more than one wrap)
309 for n = 2:num_collects;
310     if  ambig_locs_crs_ss{n}(1) > ambig_locs_crs_ss{n-1}(1)
311         shift_req = shift_req + 1;
312     end
313 end
314 if shift_req ~= 0;
315    ambig_locs_crs_ss{n} = circshift(ambig_locs_crs_ss{n},[0,shift_req]);
316 end
317 
318 % MUSIC plot
319 fignum(1) = figure();
320 plot(sinespace,10*log10(abs(music_spectrum)))
321 grid on
322 xlabel('Angle from boresight (sinespace)','fontsize',14)
323 ylabel('Orthogonality (dB)','fontsize',14)
324 set(gca,'xtick',-1:.2:1)
325 if num_collects > 1
326     legend('c_1','c_2','c_3','c_4','c_5','c_6','c_7','c_8')
327 end
328 title('MUSIC Specta','fontsize',14)
329 
330 % Signal FFT plot
331 fignum(5) = figure();
332 SIG = fft(sigs.');
333 stem(df/1e6,abs(SIG))
334 xlim([fmin fmax]/1e6);set(gca,'xtick',(fmin:1e8:fmax)/1e6);
```

```
grid on
xlabel('frequency (MHz)','fontsize',14)
ylabel('Volts (V)','fontsize',14)
legend(sig_names)
title('Transmitted signals','fontsize',14)

% plot locations and estimated DOA lines
% signals are different colors and ambiguities are the same color
cnum = num_collects;
fignum(2) = figure();
plot(e_pos(1,:),e_pos(2,:),'k+','markersize',10)
hold on
plot(coll_pos(1,1:cnum),coll_pos(2,1:cnum),'ko','markersize',8)
xlim([-250 250])
ylim([-25 300])
x1(1,1,:) = coll_pos(1,:);
x1 = repmat(x1, [1 q_est 1]);
y1(1,1,:) = coll_pos(2,:);
y1 = repmat(y1, [1 q_est 1]);
for n = 1:cnum
    for m = 1:num_lobes;
        % w0 = {# coll}(# ambig, # sig, sin(DOA)|cos(DOA))
        w0{n} = doa_est_ss{n};
        w0{n}(:,:,2) = sqrt(1-w0{n}(:,:,1).^2);
        endpt{n}(m,1:size(w0{n},2),:) = w0{n}(m,:,:)*300 + ...
            repmat(reshape(coll_pos(:,n),1,1,2),[1,q_est,1]);
        % plot initial MUSIC DOAs
        line([repmat(coll_pos(1,n),1,q_est); endpt{n}(m,:,1)],...
            [repmat(coll_pos(2,n),1,q_est);endpt{n}(m,:,2)],...
            'linestyle','-')
        if n > 1 && num_lobes ~=1;
            plot(intersect_pos{n}(m,:,1),intersect_pos{n}(m,:,2),...
                'o','color',colors(n-1,:));
            if n > 2
                plot(mid_pos{n}(m,:,1),mid_pos{n}(m,:,2), 'ks');
            end
        end
    end
end
```

```
xlabel('In-track Range (nmi)','fontsize',14)
ylabel('Cross-track Range (nmi)','fontsize',14)
title('Map View','fontsize',14)

% plot locations and estimated DOA lines for frequency fixed DOAs
% signals are different colors and ambiguities are the same color
fignum(3) = figure();
plot(e_pos(1,:),e_pos(2,:),'k+','markersize',10)
hold on
plot(coll_pos(1,:),coll_pos(2,:),'ko','markersize',8)
xlim([-250 250])
ylim([-25 300])
for n = 1:num_collects
    for m = 1:num_lobes;
        % plot initial MUSIC DOAs
        line([repmat(coll_pos(1,n),1,q_est); endpt{n}(m,:,1)],...
            [repmat(coll_pos(2,n),1,q_est);endpt{n}(m,:,2)],...
            'linestyle','--')
        if n > 1 && num_lobes ~=1;
            plot(intersect_pos{n}(m,:,1),intersect_pos{n}(m,:,2),...
                'o','color',colors(n,:));
            if n > 2
                plot(mid_pos{n}(m,:,1),mid_pos{n}(m,:,2), 'ks');
            end
        end
        endpt_ff{n}(m,1:size(w{n},2),:) = w{n}(m,:,:)*300 + ...
            repmat(reshape(coll_pos(:,n),1,1,2),[1,q_est,1]);
        %plot frequency corrected DOAs
        line([repmat(coll_pos(1,n),1,q_est); endpt_ff{n}(m,:,1)],...
            [repmat(coll_pos(2,n),1,q_est);endpt_ff{n}(m,:,2)]);
    end
end
if exist('e_pos_est')
    plot(e_pos_est(1,:),e_pos_est(2,:),'kd','markersize',8)
end
xlabel('In-track Range (nmi)','fontsize',14)
ylabel('Cross-track Range (nmi)','fontsize',14)
title('Map View (frequecy fixed)','fontsize',14)

```

```
% Transmitted signal plot
fignum(4) = figure();
colors = get(gca,'ColorOrder');
for n = 1:q
    subplot(q,1,n)
        plot(t/1e-9,real(sigs(n,:)),'color',colors(n,:))
        if n < q
            set(gca,'xtick',[])
        end
        xlim([0 40])
        legend(sig_names{n})
end
xlabel('time (ns)','fontsize',14)
ax = axes('Units','Normal','Visible','off');
set(get(ax,'Ylabel'),'visible','on')
ylabel('Volts (V)','fontsize',14)
title('Transmitted signals','fontsize',14)
set(get(ax,'title'),'visible','on')

% Received signal plot
fignum(6) = figure();
for n = 1:p
    subplot(p,1,n)
        plot(t/1e-9,real(X(n,:)))
        if n < p
            set(gca,'xtick',[])
        end
        xlim([0 40])
        legend(strcat('r_',num2str(n)))
end
xlabel('time (ns)','fontsize',14)
ax = axes('Units','Normal','Visible','off');
set(get(ax,'Ylabel'),'visible','on')
ylabel('Volts (V)','fontsize',14)
title('Received signals','fontsize',14)
set(get(ax,'title'),'visible','on')

% FFT of received signal
fignum(7) = figure();
```

```
RSIG = fft(X(1,:));
stem(df/1e6,abs(RSIG))
xlim([fmin fmax]/1e6);set(gca,'xtick',(fmin:1e8:fmax)/1e6);
grid on
xlabel('frequency (MHz)','fontsize',14)
ylabel('Volts (V)','fontsize',14)
title('Received signal','fontsize',14)

% Reconstructed signal plot
fignum(8) = figure();
for n = 1:q_est
    subplot(q_est,1,n)
        plot(t/1e-9,real(sig_est{1}(n,:)),'color',colors(n,:))
        if n < q
            set(gca,'xtick',[])
        end
        xlim([0 40])
        legend(strcat('s_',num2str(n)))
end
xlabel('time (s)','fontsize',14)
ax = axes('Units','Normal','Visible','off');
set(get(ax,'Ylabel'),'visible','on')
ylabel('Volts (V)','fontsize',14)
set(get(ax,'Ylabel'),'visible','on')
title('Reconstructed signals','fontsize',14)
set(get(ax,'title'),'visible','on')

% Reconstructed Signal FFT plot
fignum(9) = figure();
stem(df/1e6,abs(SIG_est{1}.'))
xlim([fmin fmax]/1e6);set(gca,'xtick',(fmin:1e8:fmax)/1e6);
grid on
xlabel('frequency (MHz)','fontsize',14)
legend('s_1','s_2','s_3','s_4')
ylabel('Volts (V)','fontsize',14)
title('Reconstructed signals','fontsize',14)

% Array beamshape plot at desired angle (0 for boresight)
[~, indx] = min(abs(angles-0));
```

```
C0 = v(:,indx)'*v;
figure()
plot(sinespace,20*log10(abs(C0)))
ylim([-20 0])
title('Array Pattern','fontsize',14)
xlabel('sinespace')
ylabel('Normalized gain (dB)')
```

B.2 Signal Configuration Example Script

```
%% Signal Config Example
%
% This script defines receiver and signal parameters for use with the
% main MUSIC direction finding and geolocation script and provides
% example definitions.
%
% Receiver parameters (recParam.xxx):
%   - centerfreq - array center frequency in Hz
%   - sampfreq - sampling frequency in Hz (keep in mind top end of
%   desired frequency range)
%   - dur - signal duration (receiver duration will be slightly larger
%   and will depend on array baseline)
%
% Signal parameters (sigParam{n}.xxx):
%   - All signals require:
%       - f - signal center frequency
%       - type - options are 'sin', 'QPSK', 'LFM'
%       - pos - location in the coordinate system.  Should be positive y
%       - snr - signal power (in dB); noise power is set to 0 dB if
%       noise is enabled, so this effectively also sets the SNR
%       - name - for plot labeling
%   - LFM also requires:
%       - B - signal bandwidth in Hz
%   - QPSK also requires:
%       - B - OPTIONAL, use if defining code as random sequence in order
%       to obtain appropriate length
%       - code - integer (0-3) based coding vector for QPSK phases.
%       Bandwidth is based on code length: sigParam.B = K/recParam.dur,
```

```
%       where K is the number of subcodes in the vector.
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% version name useful for specifying particular plots
sigversion = 'sig_version';

% receiver parameters
recParam.centerfreq = 1e9;  % Hz
recParam.sampfreq = 4e9;    % Hz
recParam.dur = 100e-9;      % s

% signal definitions
sigParam{1}.f = 1000e6;
sigParam{1}.B = 50e6;
sigParam{1}.type = 'QPSK';
sigParam{1}.code = randi([0 3],floor(sigParam{1}.B*recParam.dur),1);
sigParam{1}.pos = [sind(15); cosd(15)]*110;
sigParam{1}.snr = 10;
sigParam{1}.name = strcat(num2str(sigParam{1}.f/1e6),' MHz QPSK');

sigParam{2}.f = 1105e6;
sigParam{2}.type = 'sin';
sigParam{2}.pos = [90; 120];
sigParam{2}.snr = 10;
sigParam{2}.name = strcat(num2str(sigParam{2}.f/1e6),' MHz sine');

sigParam{3}.f = 1050e6;
sigParam{3}.B = 100e6;
sigParam{3}.type = 'LFM';
sigParam{3}.pos = [sind(-32); cosd(-32)]*150;
sigParam{3}.snr = 10;
sigParam{3}.name = strcat(num2str((sigParam{3}.f-sigParam{3}.B/2)/1e6)...
    ,'-', num2str((sigParam{3}.f+sigParam{3}.B/2)/1e6), ' MHz LFM');
```

B.3 Interferometric Signal Generation Function

```
function [sigs, X, e_pos, sig_names, doas, N, t] =...
    interferometry_sig_gen(sigParam, recParam, selected_sigs,...
    coll_pos, coll_vel, rec_array, noiseon)
%
% Purpose:
%   Generate signals at receiver based on pre-defined signal parameters,
% receiver array layout, collector position, and emitter position.
% Useful for subspace-based direction finding algorithms.
%
% Inputs:
%   - sigParam - structure of signal parameters (see example file)
%   - recParam - structure of receiver parameters (see example file)
%   - selected_sigs - vector where a specific subset of signals from
%   sigParam can be specified. e.g. [1 3]
%   - coll_pos - collector position matrix in [x;y], subsequent
%   positions in subsequent columns
%   - coll_vel - collector velocity in [x;y].  Assumed constant for all
%   collections.
%   - rec_array - linear receiver element location definition; should
%   specify receiver locations as distance along a line from the origin
%   - noiseon - equals 0 if additive noise is disabled and 1 if AWGN is
%   enabled
%
% Outputs:
%   - sigs - q x N vector of source signals (q is number of signals, N
%   is the number of samples
%   - X - p x N vector of received signals (p is number of receivers)
%   - e_pos - emitter positions in [x;y], subsequent emitter positions
%   in subsequent columns
%   - sig_names - cell array of signal names
%   - doas - signal DOAs in degrees. c x q matrix (c is the number
%   of collections)
%   - N - number of samples in collection
%   - t - time vector for collection (assumed to start at time zero)
%
% Custom functions called:
```

```
%   - gen_sin - generate phase shifted sinusoidal signal at receivers
%   - gen_qpsk - generate phase shifted QPSK signal at receivers
%   - gen_chirp - generate phase shifted LFM signal at receivers
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global c
p = length(rec_array);  % number of receivers
q = length(selected_sigs); % number of signals

%% Individula signal generation

b = 0;
for m = selected_sigs;  % cycle through desired signals
    b = b + 1;  % counter
    switch sigParam{m}.type
        case 'sin'
            [sig{b}, ~, doas(b), ~] = gen_sin(sigParam{m},...
                recParam, coll_pos, coll_vel, rec_array);
        case 'QPSK'
            [sig{b}, ~, doas(b), ~] = gen_qpsk(sigParam{m},...
                recParam, coll_pos, coll_vel, rec_array);
        case 'LFM'
            [sig{b}, ~, doas(b), ~] = gen_chirp(sigParam{m},...
                recParam, coll_pos, coll_vel, rec_array);
    end
    e_pos(:,b) = sigParam{m}.pos;   % emitter positions
    sig_names{b} = sigParam{m}.name; % signal names

    % adjust signal power
    sig{b} = sig{b}*sqrt(10^(sigParam{m}.snr/10));
end

% number of samples; created to capture full signal length regardless of
% DOA (includes additional time for 90 deg signal to traverse the array)
```

```
N = ceil((recParam.dur+(max(rec_array)*sind(90)./c))*recParam.sampfreq);
t = (0:N-1)/recParam.sampfreq; % time vector

%% Signal combination at array

% initializations
sigs = zeros(q,N);
Xo = zeros(p,N);

% create transmitted signal matrix and received signal matrix
for m = 1:q
    sigs(m,1:size(sig{m},2)) = sig{m}(1,:);
    Xo(:,1:size(sig{m},2)) = Xo(:,1:size(sig{m},2)) + sig{m};
end
Xa = Xo; % for future use (attenuation)

%% Additive Noise

% receiver noise variance (assume same for all elements)
noise_var = 1;  % normalized noise

if noiseon;
    rec_noise = sqrt(noise_var/2)*(randn(p,N) + 1j*randn(p,N)); % AWGN
else
    rec_noise = 0; % no noise case
end
X = Xa + rec_noise; % received signal matrix with receiver noise
```

B.4 DOA and Time Delay Calculation Function

```
function [doa, t_d] = doa_calc(sigParam, c_pos, c_vel, rec_array)
%
% Purpose:
% Determine emitter DOA and delay vector which defines when the signal
% (assumed plane wave) arrives at a receiver
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL              NOTE
```

```
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global c

% if no velocity, assume platform facing +x
if norm(c_vel) == 0;
    c_vel = [1;0];
end

e_pos = sigParam.pos; % emitter positions
e_dist = norm(e_pos-c_pos); % distance to emitter

% emitter DOA relative to array broadside
doa = 90 - acosd((c_vel'*(e_pos-c_pos))./(e_dist*norm(c_vel)));

% delay vector (column)
t_d = rec_array*sind(doa)/c;
```

B.5 Sinusoid Generation Function

```
function [s, t_rec, doa, t_d] =...
    gen_sin(sigParam, recParam, c_pos, c_vel, rec_array)
%
% Purpose:
% Generate sinusoidal pulse at receivers.
%
% Custom functions called:
%   - doa_calc - calculates signal DOA and delay vector
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

f = sigParam.f; % center frequency
```

```
ts = 1/recParam.sampfreq; % sample duration
T = recParam.dur;   % signal duration

% calculate DOA and delay vector (column)
[doa, t_d] = doa_calc(sigParam, c_pos, c_vel, rec_array);

% collection time matrix, begins at time the first receiver receives
% beginning of signal until the last receivers receives end of signal.
% Signal at origin is time zero
t_coll = repmat((ceil(min(t_d)/ts)*ts:ts:T-ts+max(t_d)),...
    [length(t_d),1]);

% function to calculate delay at each each receiver
t_ref = @(t)t - repmat(t_d,[1,length(t_coll)]);
% matrix of shifted signal arrival times per receiver
t_rec = t_ref(t_coll);

% generate complex analytic sine signal
s = exp(1j*(2*pi*f*t_rec-pi/2));
% set portions outside of pulse to zero
s(t_rec < 0) = 0;
s(t_rec > T-ts) = 0;
```

B.6 QPSK Generation Function

```
function [s, t_rec, doa, t_d] =...
    gen_qpsk(sigParam, recParam, c_pos, c_vel, rec_array)
%
% Purpose:
% Generate QPSK pulse at receivers.
%
% Custom functions called:
%   - doa_calc - calculates signal DOA and delay vector
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

f = sigParam.f; % center frequency
ts = 1/recParam.sampfreq; % sample duration
T = recParam.dur;   % signal duration
z = sigParam.code; % code
N = round(T/ts); % number of samples

% determine appropriate subcode value for each sample
z_long = repmat(z,[1 ceil(T/(length(z)*ts))])';
z_long = z_long(:);
z_long = z_long(1:N);

% calculate DOA and delay vector (column)
[doa, t_d] = doa_calc(sigParam, c_pos, c_vel, rec_array);

% force signal to exist in positive time
if min(t_d) < 0
    t_d = t_d - min(t_d);
end

% convert code integers to phase
phi_opt = [pi/4, 3*pi/4, 5*pi/4, 7*pi/4];
phi = phi_opt(z_long+1);

% collection time matrix, begins at time the first receiver receives
% beginning of signal until the last receivers receives end of signal.
% Signal at origin is time zero
t_coll = repmat((ceil(min(t_d)/ts)*ts:ts:T-ts+max(t_d)),...
    [length(t_d),1]);

% function to calculate delay at each each receiver
t_ref = @(t)t - repmat(t_d,[1,length(t_coll)]);
% matrix of shifted signal arrival times per receiver
t_rec = t_ref(t_coll);

% generate complex analytic signal
s = exp(1j*(2*pi*f*t_rec));
% set portions outside of pulse to zero
```

```
s(t_rec < 0) = 0;
s(t_rec > T-ts) = 0;

% apply phase shifts to signal
for m = 1:length(t_d)
    phaseshift = [zeros(1,abs(ceil(min(t_d)/ts)-ceil(t_d(m)/ts)))...
        exp(-1j*phi)...
        zeros(1,ceil(max(t_d)/ts)-ceil(t_d(m)/ts))];
    s(m,:) = s(m,:).*phaseshift(1:length(s));
end
```

B.7 LFM Generation Function

```
function [s, t_rec, doa, t_d] =...
    gen_chirp(sigParam, recParam, c_pos, c_vel, rec_array)
%
% Purpose:
% Generate LFM pulse at receivers.
%
% Custom functions called:
%   - doa_calc - calculates signal DOA and delay vector
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

f = sigParam.f; % center frequency
ts = 1/recParam.sampfreq; % sample duration
T = recParam.dur;   % signal duration
B = sigParam.B; % signal bandwidth

% calculate DOA and delay vector (column)
[doa, t_d] = doa_calc(sigParam, c_pos, c_vel, rec_array);

% collection time matrix, begins at time the first receiver receives
% beginning of signal until the last receivers receives end of signal.
```

```
% Signal at origin is time zero
t_coll = repmat((ceil(min(t_d)/ts)*ts:ts:T-ts+max(t_d)),...
    [length(t_d),1]);

% function to calculate delay at each each receiver
t_ref = @(t)t - repmat(t_d,[1,length(t_coll)]);
% matrix of shifted signal arrival times per receiver
t_rec = t_ref(t_coll);

% generate chirp phases
phase = (f - B/2).*(t_rec) + B/(2*T).*(t_rec).^2;

% generate signal
s = exp(1j*2*pi.*phase);
% set portions outside of pulse to zero
s(t_rec < 0) = 0;
s(t_rec > T-ts) = 0;
```

B.8 Subspace Calculation Function

```
function [q_est, EigVec_s, EigVec_n, EigVal] = subspace_calc(Xf,...
    sig_num_est, q, n)
%
% Purpose:
%   Calculate array covariance matrix (in frequency domain) and conduct
% eigenvector decomposition of said matrix.  Contains option to estimate
% number of signals via Wax's MDL or Nadakuti and Edelman's NEW method,
% or to force known number of signals
%
% Inputs:
%   - Xf - frequency domain version of received signal matrix
%   - sig_num_est - user parameter which defines how to estimate signal.
%   Options are 'none', 'MDL', or 'NEW'
%   - q - number of generated source signals
%   - n - collection number (used in alert message)
%
% Outputs:
%   - q_est - estimated number of source signals
```

```
%   - EigVec_s - signal eigenvectors
%   - EigVec_n - noise eigenvectors
%   - EigVal - eigenvalues
%
% Custom functions called:
%   - num_sig_est_MDL - Wax's MDL estimator (1985)
%   - num_sig_est_NEW - Nadakuti and Edelman's estimator (2008)
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Subspace determination

[p,N] = size(Xf);

Rxx = Xf*Xf'/N;                 % Array covariance matrix
[EigVec ,EigVal] = eig(Rxx);    % eigendecomposition

% Sort eigenvalues, then rearrange eigenvectors to match
[EigVal,indx] = sort(diag(EigVal), 1, 'descend');
EigVec = EigVec (:,indx);

%% Signal Number Estimation

switch sig_num_est
    case 'none'
        q_est = q;
    case 'MDL'
        % use Wax's MDL method
        q_est = num_sig_est_MDL(p, N, EigVal);
    case 'NEW'
        % use Nadakuti and Edelman's method
        q_est = num_sig_est_NEW(p, N, EigVal);
    otherwise
        error('Incorrect signal number estimator specified!')
end
```

```
58
59 if q_est ~= q
60     fprintf(['Incorrect signal number estimate (%1d vice %1d)' ...
61         ' at collection number %1d.\n\n'], q_est, q, n);
62     input('Reset to known number of signals (y/n)?',reset_ans);
63
64     switch reset_ans
65         case {'y','Y','yes','Yes','YES'}
66             fprintf('Resetting to known value\n\n')
67             q_est = q;
68         otherwise
69             fprintf('Continuing without reset\n\n')
70     end
71
72 end
73
74 %% Subspace determination cont.
75
76 EigVec_s = EigVec(:,1:q_est);   % signal eigenvectors
77 EigVec_n = EigVec(:,q_est+1:p); % noise eigenvectors
```

B.9 MUSIC Calculation Function

```
1  function music_spec = music_calc(A,E_n)
2  %
3  % Purpose:
4  %   Calculate the MUSIC (Schmidt, 1986) spectrum, given a steering
5  % matrix A and the noise eigenvectors E_n of the received signal
6  % covariance matrix. The function searches through provided steering
7  % matrix to find steering vectors (i.e. angles) which are perpendicular
8  % to the noise eigenvectors. These locations indicate signal directions
9  % of arrival.
10 %
11 % Inputs:
12 %   - A - steering matrix (n x m), where n is the number of receivers
13 %   and m is the number of discrete angles for which the spectrum is
14 %   being calculated.
15 %   - E_n - noise eigenvectors. (k x k), where 0 < k < n.
```

```
%
% Outputs:
%   - music_spec - MUSIC spectrum
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% execute MUSIC search
for a = 1:size(A,2)
    music_spec(1,a)=(A(:,a)'*A(:,a))/(A(:,a)'*(E_n*E_n')*A(:,a));
end
```

B.10 2D Interferometric Geolocation Function

```
 function e_pos_est = geolocate2(coll_pos, doa_est_ss)
%
% Purpose:
% least squares solution to location based on closest point to multiple
% DOA vectors.
%
% reference:
% - http://mathworld.wolfram.com/Point-LineDistance3-Dimensional.html
% NOTE: refer to thesis text for the 2D matrix version of formula in
% MathWorld reference
%
% History:
% DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
% 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[~,c] = size(coll_pos);     % number of collections
[~,q] = size(doa_est_ss);   % number of emitters

a = coll_pos;       % collector position
```

```
22
23 % unit vector along DOA
24 d = [permute(doa_est_ss,[3 1 2]);
25     sqrt(1-(permute(doa_est_ss,[3 1 2])).^2)];
26
27 % create matrix problem for each signal (refer to thesis)
28 for m = 1:q
29     e = (d(:,:,m).^2);
30     f = d(1,:,m).*d(2,:,m);
31     A = [sum(e(2,:)), -1*sum(f);
32         -1*sum(f), sum(e(1,:))];
33
34     a_long = reshape(a,[1,2*c]);
35     b1 = reshape([e(2,:); -1*f], [1,2*c]);
36     b2 = reshape([-1*f; e(1,:)], [1,2*c]);
37
38     B = [sum(a_long.*b1); sum(a_long.*b2)];
39
40     % solve matrix problem for estimated emitter location
41     e_pos_est(:,m) = A\B;
42 end
```

B.11 Point Motion Metric Function

```
1  function motion = pt_motion(positions)
2  %
3  % Purpose:
4  % Metric for estimating which DOA is true DOA and which is ambiguity.
5  %
6  % Inputs:
7  % - positions - matrix of points (either DOA intersections or
8  % DOA mid-positions.  Format is: {coll #}( x|y|(z), sig #, lobe #)
9  %
10 % History:
11 % DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
12 % 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
13 %
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
15
16 % this allows metric to be used for either DOA intersections or
17 % intersection mid-positions
18 if isempty(positions{2})
19     first_pt = 3;   % mid-positions will start at collection 3
20 else
21     first_pt = 2;   % intersections will start at collection 2
22 end
23
24 % Change in x value from first to most recent positions
25 x_diff = positions{end}(:,:,1) - positions{first_pt}(:,:,1);
26 % Change in y value from first to most recent positions
27 y_diff = positions{end}(:,:,2) - positions{first_pt}(:,:,2);
28 % Magnitude (squared) of movement between first and last positions
29 motion = x_diff.^2 + y_diff.^2;
```

B.12 Wax's MDL Estimator Function

```
1  function q_est = num_sig_est_MDL(p,N,EigVal)
2  % Number of Signals Estimation (Wax 1985), MDL estimator
3  %
4  % p is the number of receivers
5  % N is the number of samples
6  % EigVal is the list of eigenvalues of the received signal covariance
7  % matrix, sorted from largest to smallest.
8  %
9  % History:
10 % DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
11 % 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
12 %
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 for k = 0:p-1;
16     g0 = prod(EigVal(k+1:p).^(1/(p-k)));
17     a0 = (1/(p-k)*sum(EigVal(k+1:p)));
18     free_adj_param = .5*k*(2*p-k)*log(N);
19     MDL(k+1) = -((p-k)*N)*log(g0/a0) + free_adj_param;
20 end
```

```
21
22 [~, q_est] = min(MDL);
23 q_est = q_est-1;    % correct for MATLAB 1 vice 0 index
24
25 end
```

B.13 Nadakuti and Edelman's Estimator Function

```
1  function q_est = num_sig_est_NEW(p,N,EigVal)
2  % Number of Signals Estimation  (Nadakuditi and Edelman 2008), 'new'
3  % estimator
4  %
5  % p is the number of receivers
6  % N is the number of samples
7  % EigVal is the list of eigenvalues of the received signal covariance
8  % matrix, sorted from largest to smallest.
9  %
10 % History:
11 % DATE (YYMMDD) AUTHOR/EDITOR     EMAIL             NOTE
12 % 141203        Chris Straessle   gcstraes@nps.edu  Original thesis code
13 %
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15
16 for k = 0:p-1;
17     part1 = (p-k)*sum(EigVal(k+1:p).^2)/sum(EigVal(k+1:p)).^2;
18     t_k = (part1 - (1+p/N))*p;
19     NEW(k+1) = .5*(N*t_k/p)^2 + 2*(k+1);
20 end
21 [~, q_est] = min(NEW);
22 q_est = q_est-1;    % correct for MATLAB 1 vice 0 index
23
24 end
```

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] *Department of Defense Dictionary of Military and Associated Terms*, Joint Publication 1-02, U.S. Joint Chiefs of Staff, Washington, DC, 8 Nov. 2010, (As amended through 15 Aug. 2014).

[2] R. Schmidt, "Multiple emitter location and signal parameter estimation," *Antennas and Propagation, IEEE Transactions on*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[3] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 984–995, July 1989.

[4] H. H. Loomis, Jr., "Geolocation of electromagnetic emitters," Naval Postgraduate School, Monterey, CA, NPS Technical Report NPS-EC-00-003, Jan. 2001.

[5] M. G. Price, "Mathematics of geolocation," Systeka, Inc., Seabrook, MD, Report, Feb. 1995.

[6] C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1992.

[7] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '83.*, vol. 8, Apr 1983, pp. 336–339.

[8] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 4, pp. 806–811, Aug. 1985.

[9] J. E. Evans, D. F. Sun, and J. R. Johnson, "Application of advanced signal processing techniques to angle of arrival estimation in ATC navigation and surveillance systems," MIT Lincoln Lab, Lexington, MA, Technical Report 582, June 1982.

[10] M. Doron, E. Doron, and A. Weiss, "Coherent wide-band processing for arbitrary array geometry," *IEEE Transactions on Signal Processing*, vol. 41, no. 1, pp. 414–417, Jan. 1993.

[11] B. Friedlander and A. Weiss, "Direction finding for wide-band signals using an interpolated array," *IEEE Transactions on Signal Processing*, vol. 41, no. 4, pp. 1618–1634, Apr. 1993.

[12] M. Kaveh and A. Barabell, "The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 331–341, Apr. 1986.

[13] A. Swindlehurst and T. Kailath, "A performance analysis of subspace-based methods in the presence of model errors part I: The MUSIC algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1758–1774, July 1992.

[14] S. Stein, "Algorithms for ambiguity function processing," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 29, no. 3, pp. 588–599, June 1981.

[15] E. Gönen and J. Mendel, "Subspace-based direction-finding methods," in *The Digital Signal Processing Handbook: Wireless, Networking, Radar, Sensor Array Processing, and Nonlinear Signal Processing*, 2nd ed., V. Madisetti, Ed. Boca Raton, FL: CRC Press, 2009, ch. 3.

[16] E. W. Weisstein. (2014, Apr. 28). Point-line distance–3-dimensional. From MathWorld–A Wolfram Web Resource. [Online]. Available: http://mathworld.wolfram.com/Point-LineDistance3-Dimensional.html

[17] R. Penrose, "A generalized inverse for matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, July 1955.

[18] M. I. Skolnik, *Radar Handbook*, 2nd ed. The McGraw-Hill Companies, 1990.

[19] T. T. Ha, *Theory and Design of Digital Communication Systems*. Cambridge, UK: Cambridge University Press, 2011.

[20] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 2, pp. 387–392, Apr. 1985.

[21] F. R. Hill and R. Pickholtz, "Estimating the number of signals using the eigenvalues of the correlation matrix," in *, 1989 IEEE Military Communications Conference, 1989. MILCOM '89. Conference Record. Bridging the Gap. Interoperability, Survivability, Security*, Oct. 1989, pp. 353–358 vol.2.

[22] R. Nadakuditi and A. Edelman, "Sample eigenvalue based detection of high-dimensional signals in white noise using relatively few samples," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2625–2638, July 2008.

[23] B. Friedlander and A. Weiss, "Direction finding using spatial smoothing with interpolated arrays," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 2, pp. 574–587, Apr. 1992.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California